

Jenny Kärkkäinen

SERVICE DESIGN FOR SAP FIORI DEVELOPMENT



JYVÄSKYLÄN YLIOPISTO
INFORMAATIOTEKNOLOGIAN TIEDEKUNTA
2020

TIIVISTELMÄ

Kärkkäinen, Jenny

Palvelumuotoilu ja SAP Fiori -sovelluskehitysprosessi

Jyväskylä: Jyväskylän yliopisto, 2020, 75 s.

Tietojärjestelmätiede, pro gradu -tutkielma

Ohjaaja(t): Seppänen, Ville

Tässä tutkimuksessa keskitytään palvelumuotoiluun ja SAP Fiori-sovelluskehityksen prosessiin business-sidosryhmän näkökulmasta. SAP Fiori sovellukset ovat pääkäyttöliittymä SAP:n SAP S/4HANA-järjestelmässä. Sovelluskehitysmalleja ja -lähestymistapoja on useita, ja tämän tutkimuksen kirjallisuuskatsauksessa näistä muutamia keskeisiä lähestymistapoja tarkastellaan ja verrataan palvelumuotoilun suunnitteluvetoiseen prosessiin. Palvelumuotoilu on monitieteellinen ja käytännönläheinen lähestymistapa. Kirjallisuuskatsauksen tavoite on tarkastella mitä samankaltaisuuksia ja eroja sovelluskehitysmalleilla on palvelumuotoilun lähestymistavan kanssa. Case-tutkimusosassa pyritään vastaamaan, millaisia kokemuksia case-yrityksen businesshenkilöillä on SAP Fiori -sovellusten kehittämisprosesseista ja SAP Build-työkalusta. SAP Buil -työkalua voidaan hyödyntää suunnitteluvetoisessa SAP Fiori -sovellusten kehittämisessä palvelumuotoilun menetelmiä hyödyntäen. Tämä kvalitatiivinen tapaustutkimus on toteutettu käyttämällä tutkimusmenetelmänä teemahaastatteluja, joka on kvalitatiivista, eli laadullinen, tutkimusmenetelmä. Datan analysointi suoritettiin konventionaalista sisältöanalyysimenetelmää käyttäen. Keskeisimmät tutkimustulokset ovat, että loppukäyttäjien SAP Fiori -sovelluksien käyttöönotto on koettu helpommaksi ja SAP Fiori -sovellusten nähdään tuovan paljon uusia mahdollisuuksia. Tutkimuksessa kävi ilmi, että loppukäyttäjien tulisi olla mukana kehitysprosessin aikana, mutta heillä ei ole aikaa itse tehdä prototyyppejä tai opetella käyttämään SAP Buildin kaltaista työkalua. Tietohallinnon (IM) rooli koettiin kaikista epäselvimmäksi ja tulevaisuutta varten toivottiin aktiivisempaa roolia ja tukea etenkin kehityksen alkuvaiheisiin. Tutkimuksessa kävi ilmi, että palvelumuotoilun elementeistä voisi olla hyötyä esimerkiksi yhteistyön kehittämiseksi IM:n ja busineksen välillä tapausorganisaatiossa.

Asiasanat: palvelumuotoilu, sovelluskehitysprosessi, SAP Fiori

ABSTRACT

Kärkkäinen, Jenny

Service design for SAP Fiori application development

Jyväskylä: University of Jyväskylä, 2020, 75 pp.

Information Systems, Master's Thesis

Supervisor(s): Seppänen, Ville

The focus of this research is on service design and SAP Fiori application development process from business stakeholder's point of view. SAP Fiori applications are the main user interface in SAP's system SAP S/4HANA. There are numerous models and approaches to application development and in this research few of those are examined and compared to service design approach in the literature review. Service design is interdisciplinary and very practical approach. The aim of this literature review is to examine what similarities and differences well-known application development approaches have with service design. The case study aims to answer what kind of experiences business stakeholders have of SAP Fiori development processes and SAP Build tool which can be used for design-led SAP Fiori application development. This qualitative case study is conducted by using open-ended interviews which is a qualitative research method. Data analysis is performed by utilising conventional content analysis. The main research findings were that SAP Fiori was perceived as easier to implement by users and providing a lot of possibilities. The end users are important to include in the process throughout and their feedback is valuable, but end users inter alia do not have time to start prototyping the ideas themselves or using tools such as SAP Build. During the development process the role of information management (IM) organisation was most unclear and more activity and input was hoped for the future especially for the start of the process. Some service design methods could be used for development process improvement and partially included to SAP Fiori application development process to enhance collaboration and ideation between the business and IM in the case company.

Keywords: service design, application development process, SAP Fiori

FIGURES

FIGURE 1 Overview of the research structure.....	11
FIGURE 2 Waterfall model of system development lifecycle (Cadle & Yates, 2009).....	17
FIGURE 3 Royce’s original model on how to implement large software developments and its sequential stages. The need for redesign discovered in testing stage moves the development back to requirement stage. (Royce, 1987.)	18
FIGURE 4 Five perspectives to service design (Stickdorn et al., 2018).	24
FIGURE 5 SAP Build application development process is design-led, and the process is based on service design methodology (SAP, 2020e).	29
FIGURE 6 History of SAP (SAP, 2020b).	35
FIGURE 7 Overview of SAP ERP architecture (Böder & Gröne, 2013).....	35
FIGURE 8 SAP Fiori architecture in SAP S/4HANA (SAP Help Portal, 2020)..	37
FIGURE 9 SAP Fiori applications can be used in any platform (SAP, 2020c). ...	37
FIGURE 10 SAP Fiori design principles (SAP, 2020d).	38
FIGURE 11 SAP Fiori monetary benefits for business (SAP, 2020d).	38
FIGURE 12 SAP Fiori human benefits for business (SAP, 2020d).	39
FIGURE 13 Differences between user experience and user interface according to SAP (2020d)	40
FIGURE 14 Overall enhancement request process for all SAP development and average duration of each step in the case company (Schutte, 2019).	41
FIGURE 15 Stakeholders in SAP development (Schutte, 2019).	42

TABLES

TABLE 1 Home ground for agile and plan-driven methods (Boehm, 2002; according to Abrahamsson et al., 2002, 18-19).	21
TABLE 2 Principles of service design doing (Stickdorn et al., 2018, p. 27).	26
TABLE 3 Comparison of development approaches based on literature.	32
TABLE 4 Interviewee identification information.....	49

TABLE OF CONTENTS

TIIVISTELMÄ	2
ABSTRACT	3
FIGURES	4
TABLES	5
TABLE OF CONTENTS	6
1 INTRODUCTION	8
1.1 Research questions	10
1.2 Thesis outline.....	10
2 APPLICATION DEVELOPMENT	12
2.1 Definition	12
2.1.1 Enterprise application.....	13
2.2 Application development approaches.....	14
2.2.1 Plan-driven development	16
2.2.2 Agile development.....	19
2.3 Summary.....	21
3 SERVICE DESIGN.....	22
3.1 Definition	22
3.1.1 Service design process	26
3.1.2 Design-led development	28
3.2 Conclusions of the theory.....	31
4 CASE: SAP FIORI APPLICATION DEVELOPMENT	34
4.1 Enterprise Resource Planning.....	34
4.2 SAP Fiori	36
4.3 SAP development process today	40
4.4 Summary.....	42
5 RESEARCH METHODS AND ANALYSIS	43
5.1 Choosing the methods	43
5.2 Data collection.....	45
5.3 Data analysis.....	46
6 RESULTS	49
6.1 SAP Fiori development process.....	50

6.2	SAP Fiori development experiences	51
6.3	Collaboration experiences	54
6.4	Service design and prototyping methods	57
6.5	SAP Build and design-led development	58
6.6	Future development	61
6.7	Research limitations	62
7	DISCUSSION	64
7.1	Summary	64
7.2	Research questions and key findings.....	68
	REFERENCES.....	70
	APPENDIX 1 INTERVIEW QUESTIONS	75

1 INTRODUCTION

Advances in information technologies have enabled organizations to improve the efficiency and delivery of services (Venkatesh, Thong, Chan & Hu, 2014). The arrival of new technologies has dramatically altered the way in which Information Systems (IS) are being conceived, developed and managed in organizations: Technologies such as blockchain, the Internet of Things (IoTs), and rapid automation of processes through use of artificial intelligence (AI) and machine learning (ML) has impacted every side of IS development and project management. (ICIS, 2020). Information systems are expensive to develop and maintain, and therefore businesses do not commission them just for the fun of it. The need for an information system must grow out of some perceived business requirement, and the justification for it must be expressed in business terms. And, although this is well enough understood in theory, it is surprising how often IS projects start without clear links back to the business plans and strategies: System developers often take sketchy briefs and start to develop something they think will meet the need - and are then unpleasantly surprised when the user, sponsor or the system refuses to accept it because it does not properly meet their specific business requirements. (Yeates, 2004). Combining new software development approaches with diverse software platforms and application environments provide the opportunity to broaden the array of approaches to design and development available to IS project managers and to offer the prospect of approaches better differentiated to organizational settings, personnel skills, and task demand (ICIS, 2020). Service design is a design-led development approach in which visualisation, ideation, planning, collaboration and prototyping are in essential roles for developing solutions or processes which answer to the problem or need (Mager, 2008; Stickdorn et al., 2018).

Enterprise resource planning (ERP) systems are an essential part of organizations IT functions. There are many ERP system providers and in this study ERP provider SAP SE's system is in focus. SAP is the largest ERP system provider in the world and has brought to the market a new system called SAP Business Suite 4 SAP HANA (SAP S/4HANA for short) (SAP, 2020a). In SAP S/4HANA, the SAP Fiori is the new user interface and therefore the change to

new system and the role of SAP Fiori enterprise applications is significant from end-user's perspective. Enterprise applications are an outstanding case study for the entire software industry because most of enterprise applications demonstrate all the real-world problems businesses have (Spratt, 2000).

This study is a qualitative case study and conducted according to case research method (Benbasat, Goldstein & Mead, 1987; Darke, Shanks & Broadbent (1998). The case object of this study is businesspersons in the case company. The case company is a global company in the marine and energy sector. The purpose of this research is to look for an understanding of the SAP Fiori application development phenomenon in the case company, their challenges and hopes for future, and in the discussion explore what kind of elements of service design approach could be utilized in case company's SAP Fiori application development process in the future. The research is relevant from organizational perspective because of the future transition from current ERP system, SAP ERP, to SAP S/4HANA. As many other companies are also moving to SAP S/4HANA, there is a practical need outside of case company as well. At this point in time it is relevant to study and understand the pain points and challenges in the current way of working in order to gain understanding on how the process could be enhanced for the future. A fundamental question in IS research field is how information systems can be utilized efficiently or in other good ways in human activities (Grover & Lyytinen, 2016). Service design as a relatively new approach to development process provides opportunity to broaden the range of approaches and offers possibilities of approaches to be better differentiated to organizational settings, personnel skills, and task demand (ICIS, 2020). For these reasons, the study is interesting and relevant from scientific perspective as well.

In literature review application development approaches and service design and design-led development approach are elaborated and compared. Case section of the study introduces the case phenomenon, data collection and data analysis and results. The main research findings are agile approach and service design approach have much in common compared to plan-driven approach. SAP Fiori applications were perceived as easier to implement by users compared to traditional SAP applications and it was seen as a missing piece that brings a lot of possibilities. The business end users are important to have involved throughout the development process and their input is valuable, but end users do not have time to start prototyping the ideas themselves with a tool such as SAP Build. This could be done as today in the business support functions that were the focus group for this research. During the development process the role of information management (IM) organisation was most unclear and more activity and input was hoped for the future especially for the start of the process. Service design elements could be utilized to enhance collaboration, but it would need to be investigated whether there are enough resources in IM for more active role.

1.1 Research questions

This study aims to answer in literature review to the following research questions:

1. What kind of differences there are between well-known application development approaches and service design approach?

With the case study, the research aims to answer following questions:

1. What kind of SAP Fiori development and SAP Build tool experiences business has and what kind of hopes there are for future development?
2. What kind of elements of service design approach could be integrated to SAP Fiori development process to prevent the challenges and address the future wishes?

1.2 Thesis outline

The research is conducted as a qualitative case research. To judge the appropriateness of the case strategy and if the case method is a useful approach to answer these research questions, we answer the following questions according to Benbasat et al. (1987):

1. Can the phenomenon of interest be studied outside its natural setting?
2. Must the study focus on contemporary events?
3. Is control or manipulation of subjects or events necessary?
4. Does the phenomenon of interest enjoy an established theoretical base?

The case phenomenon is SAP Fiori application development process in the case company. The phenomenon of interest can be studied outside its natural setting as there are many similar sized organizations using SAP ERP system moving to SAP S/4HANA and implementing SAP Fiori user interface and applications. Also, there are probably similar challenges in other organizations as well and goals for improved development process. The study focuses on contemporary events. There is an established theoretical base for application development processes.

The structure of this study is following (see figure 1): Introduction is followed by a literature review of main concepts of this study: Application development and service design. First chapter focuses on defining application development, two of its development process approaches, plan-driven and agile development approaches. This is followed by defining service design and its principles, design-led development process and methods used in the

process. After defining these main concepts of this research, a comparison and analysis of the challenges and similarities of these concepts is conducted. The goal is to identify the similarities and differences between these different approaches to development process and therefore answer the research question: How traditional application development approaches, and service design approach differ from each other? This is done to analyse if the challenges in application development processes could be improved by applying elements of service design approach. The literature review ends to a summary and conclusions of the theory. Third chapter consists of case study introduction. Case company is introduced shortly and as a part of the case chapter, enterprise resource planning (ERP) system are defined and ERP system providers and products that are relevant to the case research are introduced: SAP, SAP S/4HANA, and SAP Fiori. The fourth chapter describes the research methods, data collection and data analysis processes. For data analysis, conventional content analysis is applied according to Hsieh and Shannon (2005). Data analysis is followed by results and the research ends with discussion and key findings and proposal for future research.

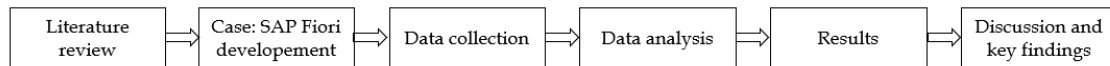


FIGURE 1 Overview of the research structure.

2 APPLICATION DEVELOPMENT

This chapter consists of theory about application development. The focus of this research is on enterprise applications development. First in this chapter, application and application development are defined. Second part of this chapter discusses the application development process models waterfall, agile, and defines the distinctions between these two, the challenges and advantages of the approaches. The chapter ends with summary.

2.1 Definition

Application software (application for short) is a system for collecting, saving, processing, and presenting data by means of a computer, and a coherent collection of automated procedures and data supporting a business objective, (ISO/IEC/IEEE 24765:2017). Application is software that is designed for users and to fulfil specific needs of a user to help them perform particular tasks or handle particular types of problems, as distinct from software that controls the computer itself. It can be also defined as software or a program that is specific to the solution of an application problem. (ISO 44001:2017, 2017). The term application is generally used when referring to a component of a software that can be executed. It consists of one or more components, modules, or subsystems. Application development is a process in which application is being developed from planning to implementation. Process is set of interrelated or interacting activities which transforms inputs into outputs. (ISO 44001:2017, 2017.).

There are many kinds of applications that can be used in different platforms. For example, mobile platforms have on top of traditional functionalities different kinds of embedded sensors such as digital compass, accelerometer, navigator, microphone, and camera (Lane et al., 2010). Portability, continuous data streaming, advanced computing power and easy dissemination of applications give them an edge over other forms of information and communication technologies (Langrial et al., 2012). This technology has been used with SAP Fiori applications which can be used in any

platform compared to SAP GUI which can be used on computer only. New technology enables working in different conditions and also new possibilities that can be done with enterprise applications. The focus of this research is on enterprise applications where traditionally work is mainly done by using a laptop or computer, but application technology and embedded sensors make it possible to expand the work to mobile devices and platforms and allow new way of working.

2.1.1 Enterprise application

Conallen (1999) defines a web application as a software system with business state where its “front end” is in large part delivered via web system: Web application is web system (web server, network, HTTP, browser) where user input (navigation and data input) affects the state of the business. Enterprise applications can be used via web browser in many platforms or devices, and the so-called enterprise resource planning (ERP) applications market was one of the fastest growing and most profitable areas of the software industry during the last three years of the 1990s (Spratt, 2000). More recently, as moving into the post-digital era, outdated technology such as legacy systems are becoming increasingly burdensome and because of this, more and more organisations are replacing their old systems with modern ERP systems and gaining competitive advantage (Panorama Consulting Group, 2020). The percentage of organisations changing from legacy systems to modern ERP system has significantly increased: This year 35 % of ERP report respondents are moving away from legacy systems whereas compared to last year, 2019, 14% of organisations were moving (Panorama Consulting Group, 2020). As most of enterprise applications demonstrate all the real-world problems that businesses have, enterprise applications are an outstanding case study for the entire software industry according to Spratt (2000). Challenges that businesses have are such as legacy applications that cannot be rewritten, monolithic code not built for easy maintenance, multiple design and execution technologies that need to be integrated, demand for new technology support and customers that won't wait years for a solution. (Spratt, 2000). According to McKeen and Smith (2002) as well, it is common in most organisations to have multiple applications (custom, legacy, and packaged), multiple platforms, multiple databases, multiple transaction processors, multiple data entry points, multiple versions of the same data, and incompatible business data.

Because of the challenges, replacement of an old application entirely with a new one has been found often easier than continuing to throw good money into aged legacy applications. But, there has been also discernible a strong desire in many organizations instead of developing custom solutions, especially for transactional applications, to acquire functionality. Enterprise application providers such as SAP, Peoplesoft, Oracle, Baan, JD Edwards, and many others have been investing heavily to upgrade the architecture of their applications over the years. Some of the packaged application vendors have acquired a poor

reputation for being better known for the time and cost involved in implementation than the resulting business benefits. In this research the focus is on enterprise application provider SAP and its SAP Fiori applications. (Spratt, 2000, 63).

Enterprise application frameworks are the cornerstone of enterprise business activities (Fayad & Hamu, 1997; according to Fayad & Schmidt, 1997) and address broad application domains, such as telecommunications, avionics, manufacturing, and financial engineering (Birrner, 1993; Laitinen, Fayad, Schmidt & Johnson, 1999; Fayad, Schmidt & Johnson, 1997; according to Fayad & Schmidt, 1997). Enterprise frameworks are expensive to develop and/or purchase but can provide a substantial return on investment: They support the development of end-user applications and products directly. In contrast, system infrastructure and middleware integration frameworks focus largely on internal software development concerns, and although these frameworks are essential to create high-quality software rapidly, they typically do not generate substantial revenue for large enterprises. This results that it is often more cost-effective to buy system infrastructure and middleware integration frameworks rather than build them in-house. (Fayad & Hamu, 1997; Laitinen et al., 1999; according to Fayad & Schmidt, 1997.).

2.2 Application development approaches

Applications such as enterprise applications are developed usually according to some development model. There are many approaches that have been used, and are being used, to develop information systems. Most structured approaches make use of a plan-driven waterfall model but the spiral, iterative, approach forms the basis of agile development (Cadle & Yates, 2009, 88). The implemented development model can be crucial for the application development success as according to Conallen (1999, 65):

Models help us understand the system by simplifying some of the details. The choice of what to model has an enormous effect on the understanding of the problem and the shape of the solution [1, p. 8]. Web applications, like other software-intensive systems, are typically represented by a set of models: use case model, implementation model, deployment model, security model, and so forth. An additional model used exclusively by Web systems is the site map, an abstraction of the Web pages and navigation routes throughout the system. (Conallen, 1999, 65.)

As stated already, there are many different models and frameworks for application development. In this chapter the following approaches are explored: plan-driven and agile development approaches. The principles, strengths and challenges of each development models are defined, and the approaches are lastly compared to each other. Prior to the exploration of these, the basic phases of development process are defined, as these development approaches use quite similar terminology even though the approaches to

development differ. Overall, development process refers to specification, construction, testing and delivery of a new application or of a discrete addition to an existing application (ISO/IEC/IEEE 90003:2018). Development can be executed in-house or outsourced to a service provider. Service provider is an organization that manages and delivers a service or services to the customer. A customer can be internal or external to the service provider's organization. (ISO/IEC/IEEE 24765:2017.).

Software development process is a process by which user needs are translated into a software product, to an enterprise application in this case. The development process involves translating user needs into software requirements, transforming the software requirements into design, implementing the design in code, testing the code, and sometimes, installing and checking out the software for operational use. These activities can overlap or be performed sequent or iteratively. (ISO/IEC/IEEE 24765:2017). Development process consist usually of scope definition, requirement gathering, design, testing, and deployment. Next these steps are defined more accurately. First, defining the scope means the process of developing a detailed description of the project and product (ISO/IEC/IEEE 90003:2018).

After the scope is clear, the requirements for the application are gathered and mapped. Requirement can be defined as a statement that translates or expresses a need and its associated constraints and conditions, or as a condition or capability that must be met or possessed by a system, system component, product, or service to satisfy an agreement, standard, specification, or other formally imposed documents. Requirements express the needs in high-level form and exists at different levels. Requirements provide value when delivered, satisfied, or met, and they include the quantified and documented needs, wants, and expectations of the sponsor, customer, and other stakeholders. Stakeholder is defined as an individual, group or team, or organization that 1) has a right, share, claim, or interest in a system or in its possession of characteristics that meet their needs and expectations, and 2) who may affect, be affected by, or perceive itself to be affected by a decision, activity, or outcome of a project. Usually for example end users can be involved as they know first-handed what kind of functionalities are essential. In this study stakeholders are business organisation, information management organisation, externals, and business end-users. (ISO/IEC/IEEE 24765:2017.).

Requirements mapping is usually followed by design phase. Design phase is not to be confused with design process. From systems and software engineering perspective, design process means defining the architecture, system elements, interfaces, and other characteristics of a system or system element. Design phase on the other hand is a period in the software life cycle during which definitions or designs for architecture, software components, interfaces, and data are created, documented, and verified to satisfy requirements. After design phase the design goes to development, where the solution is executed. The executed development is then tested. Test phase is a period of time in the software life cycle during which the components of a

software product are evaluated and integrated, and the software product is evaluated to determine whether or not requirements have been satisfied or a specific instantiation of test sub-process. Testing is activity in which a system or component is executed under specified conditions, the results are observed or recorded, and an evaluation is made of some aspect of the system or component. If testing is successful, the application goes to production and is deployed: Deployment is a phase of a project in which a system is put into operation and cutover issues are resolved. (ISO/IEC/IEEE 24765:2017.).

Many organisations these days are interested and have invested in implementing more of an agile way of working instead of plan-driven approach. Transformation from plan-driven to agile methods is a change process that needs careful planning, implementation and anchoring to organization's development culture. (Martiin, 2020). Every project, application development project as well, is constrained in different ways by its scope, time, and cost (Atkinson, 1999). McConnell (1996) described the most common mistakes related to software development which are still today relevant. These mistakes are divided into three groups: people, process, and product and technology related, and are listed almost as a guideline to what not to do in order to prevent challenges in the development project, and include mistakes such as "lack of user input", "inadequate design" and "insufficient planning" (McConnell, 1996).

2.2.1 Plan-driven development

A plan-driven development model was first introduced by Royce (1987) to be implemented in large software development projects. In this model that is today know as a waterfall model (see figure 2), system development is broken down into a number of sequential sections or stages represented by boxes, with each stage being completed before work starts on the following one (Royce, 1987). It is a plan-driven process where the output of previous stage is used as an input in the next phase. (Cadle & Yates, 2009.) The output is confirmed and accepted before using it as an input. In each stage there is two elements where the first covers the actual work being carried out in the stage. The second part is for the 'verification and validation' of that work. (Cadle & Yates, 2009.) Verification is meant for establishing the correspondence between a product and its specification and making sure are the product is built in the right way. Validation is concerned with whether the product is fit for its operational mission and are we building the right product. Usually, there is a degree of iteration of work and products within a stage but very little between stages. (Cadle & Yates, 2009.)

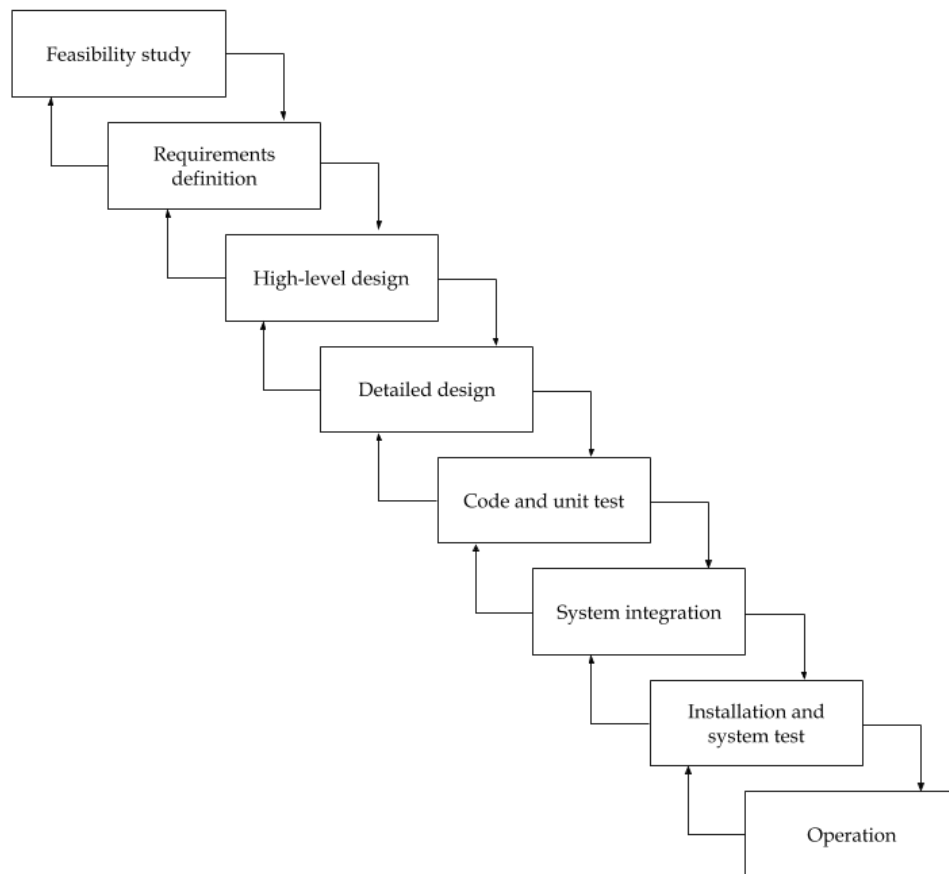


FIGURE 2 Waterfall model of system development lifecycle (Cadle & Yates, 2009).

Rework, where necessary, is carried out in succeeding stages and the original stage in which the product was produced is not revisited. If new requirements are needed, the rework is usually taken as a part of the current stage instead of going back to requirement stage. (Cadle & Yates, 2009.) However, initially Royce (1987) presented that if new requirements come up during testing, the development would go back to the requirement stage (see figure 3).

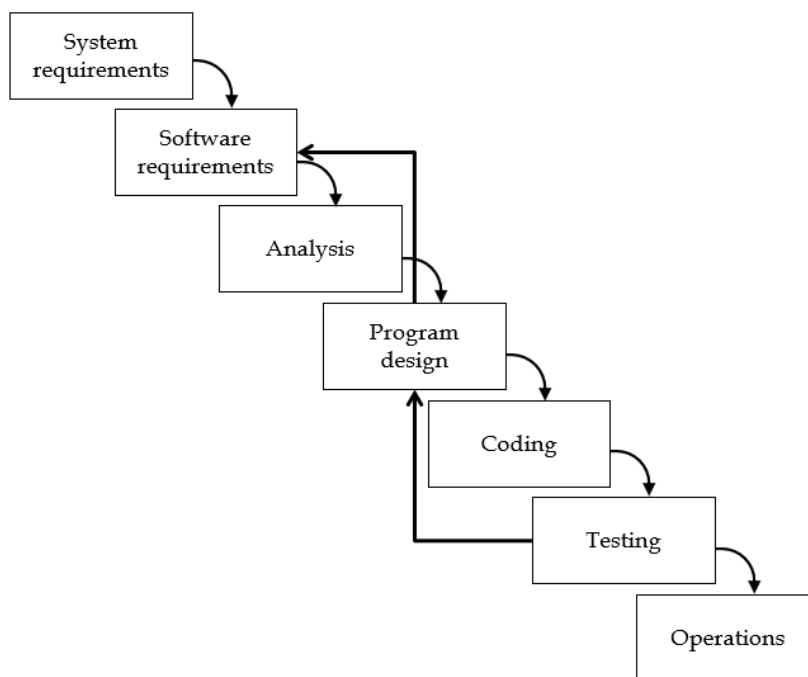


FIGURE 3 Royce's original model on how to implement large software developments and its sequential stages. The need for redesign discovered in testing stage moves the development back to requirement stage. (Royce, 1987.)

According to Cadle and Yates (2009) the waterfall model is generally nowadays taken to mean any sequential model divided into consecutive stages and having the attributes of the original model. The identification and naming of the stages are not fixed and can be modified to suit particular project characteristics. (Cadle & Yates, 2009). The challenges and strengths of waterfall development are addressed next. According to McConnell (1996, 138) waterfall model has following challenges:

- Requires accurate planning and excess documentation
- Expensive to back up
- Inflexibility
- Few visible signs of progress
- Results available only in the end. (McConnell, 1996, 138).

According to Royce (1987) as well, the waterfall implementation model is risky and invites for failure as testing phase is occurs at the end of the development model. If the testing results are unsatisfactory then a major redesign is required which can be then expected to lead to overrun in schedule and/or costs. To eliminate most of the development risks, five additional features can be added to the basic approach (Royce, 1987):

1. Program design comes first.
2. Document the design.

3. Do it twice – the first result provides an early simulation of the final product. This can be also seen in figure 2, where first a high-level design step is done, and followed by detailed design step.
4. Plan, control, and monitor testing.
5. Involve the customer. (Royce, 1987, 331-335).

Royce (1987) notifies that each of these features cost some additional money but in his experience the costs exceed the need to keep the budget smaller, as the simpler development method has not worked on large software development efforts.

2.2.2 Agile development

In this paragraph agile development model is defined. Agility is all about trusting in one's ability to respond to unpredictable events more than trusting in one's ability to plan for them (Fowler & Highsmith, 2001). Agile is a generic term applied to various systems development approaches, such as DevOps, XP, Scrum, that emphasize flexibility, speed and user involvement in development systems. Agile methods make use of relatively short 'timeboxes' to deliver packages of usable software. (Cadle & Yates, 2009). Agile development is a software development approach based on iterative development, frequent inspection and adaptation, and incremental deliveries, in which requirements and solutions evolve through collaboration in cross-functional teams and through continuous stakeholder feedback (ISO ISO/IEC/IEEE 24765:2017). The values of agile software development are following according to the agile manifesto (Beck, Beedle, Van Bennekum, Chockburn, Cunningham, Fowler, ... & Kern, 2001):

- Individuals and interactions over processes and tools.
- Working software over comprehensive documentation.
- Customer collaboration over contract negotiation.
- Responding to change over following a plan. (Beck et al., 2001).

According to the agile manifesto, while items on the right are valued, items on the left side are valued more. Agile software development principles are following:

- The highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- Welcome changing requirements, even late in development.
- Agile processes harness change for the customer's competitive advantage.
- Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

- Businesspeople and developers work together daily throughout the project.
- Build projects around motivated individuals.
- Give them the environment and support they need and trust them to get the job done.
- The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
- Working software is the primary measure of progress.
- Agile processes promote sustainable development. The sponsors, developers and users should be able to maintain a constant pace indefinitely.
- Continuous attention to technical excellence and good design enhances agility.
- Simplicity—the art of maximizing the amount of work not done—is essential.
- The best architectures, requirements and designs emerge from self-organizing teams.
- At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly. (Beck et al., 2001).

Agile methods have been seen more suitable for developing small systems development. The use of agile software development methodologies and close co-innovation with customers have shown to result in better products (Neale & Corkindale, 1998 according to Vosough, Walter, Rode, Hesse & Groh, 2016). In their research Abrahamsson, Salo, Ronkainen, and Warsta (2002, 93) review different agile methods and conclude that according to Highsmith and Highsmith (2002) all these methods place emphasis on six aspects: They are 1) delivering something useful, 2) have reliance on people, 3) are encouraging collaboration, 4) are promoting technical excellence, 5) are doing the simplest thing possible and 6) are being continuously adaptable.

The agile approach is particularly suitable in situations where future requirements are not known (e.g., Ambler, 2002; according to Abrahamsson et al., 2002). It appears, agile approaches do not provide added value to development project if the requirements are known and can be specified (Abrahamsson et al., 2002). Prototyping can be used in agile processes for development where the requirements are not known or specified. According to Cadle and Yates (2009) prototyping is an approach used mainly, though not exclusively, within agile development approaches, whereby prototypes of proposed functionality are built and then reviewed with the system's users. The prototypes may be thrown away once the specific questions they were designed to answer have been addressed or they may evolve into the finished system. Generally, the agile approach involves a number of techniques that enable the requirements, and subsequently the system, to be developed via a series of iterative activities usually involving the use of prototypes. The use of prototyping does not necessarily imply agile, however, as prototyping

techniques may be used in other types of development approach. Within an agile project, prototyping may be used in many ways, such as: assisting users to define and confirm requirements by demonstrating possibilities, investigating the effectiveness of novel methods of working, and testing performance implications, assisting in considering work practice (Abrahamsson et al., 2002).

Once developed, prototypes allow the system to be examined and reviewed by the users and modifications and refinements can be made quickly and easily. The prototypes then become the final delivered system. One of the major differences between agile and the more conventional structured methods is that iteration and rework are seen as being an integral part of the agile approach and not something to be avoided if possible, which is the view of most structured methods. (Cadle & Yates, 2009, 79).

2.3 Summary

In this chapter application development was defined, describing a general elements of application development process and defining enterprise applications that are in the focus of this research. Second part of this chapter presented three approaches to application development: Plan-driven, where waterfall model was introduced in more detail and Agile development. Both of these approaches were presented using the same typology: Introduction and definition, process characteristics, challenges, and redeeming features.

Boehm (2002; according to (Abrahamsson et al., 2002) analyses and compares the agile and plan- driven methodologies (see table 1).

TABLE 1 Home ground for agile and plan-driven methods (Boehm, 2002; according to Abrahamsson et al., 2002, 18-19).

Home-ground area	Agile methods	Plan-driven methods
Developers	Agile, knowledgeable, collocated, and collaborative	Plan-oriented; adequate skills; access to external knowledge
Customers	Dedicated, knowledgeable, collocated, collaborative, representative, and empowered	Access to knowledgeable, collaborative, representative, and empowered customers
Requirements	Largely emergent; rapid change	Knowable early; largely stable
Architecture	Designed for current requirements	Designed for current and foreseeable requirements
Refactoring	Inexpensive	Expensive
Size	Smaller teams and products	Larger teams and products
Primary objective	Rapid value	High assurance

Agile and plan-driven methods differ from each other in numerous ways but the main thing to consider when initiating development of an application is to assess what kind of development method is most suitable for the development in hand. Agile methods are usually implied in smaller products and teams where there is a dedicated team who can work together daily, whereas plan-driven methods are often used in larger teams and products (see table 1).

3 SERVICE DESIGN

Service design is still largely not well understood while product design and interaction design are establishing themselves as ordinary practices (Holmlid, 2007). This chapter consists of defining service design (SD) and its methods and design-led development process. As service design is not as well understood, a deeper exploration of the subject is provided as in previous chapter of more known approaches. First, we define service design and its key principles, followed by examining the service design process: the process phases are presented by using SAP Build tool development process model as a reference. Also methods used in the process are shortly described. The chapter ends with a comparison of service design and application development process approaches to discuss whether service design and application development processes could be applied together in order to improve efficiency and end-solutions in application development process. The principles of each are compared to each other to look for similarities and differences between these processes. Comparison analysis is followed by summary of the literature review.

3.1 Definition

Service means delivering value for the customer by facilitating results the customer wants to achieve. Service can also be defined as performance of activities, work, or duties, and thirdly as behaviour, triggered by an interaction, which adds value for the service users by creating, modifying, or consuming information; the changes become visible in the service provider's environment. A service is generally an intangible product that can be composed of other services. (ISO/IEC/IEEE 24765:2017). Under the service science umbrella, there are different service disciplines of which new opportunities and problems arise and these disciplines as well have different strengths and weaknesses. Two examples of these sub disciplines are service design and service management

that at first glance supplement each other well. The difference between service design and service management is following: Where service management mainly focuses on the management of existing services, service design mainly focuses on development of new and improvement of existing services. (Segelström & Holmlid, 2011). According to Patrício, Fisk, Cunha and Constantine (2011) the rapid evolution of service systems raises new challenges for service design, and consensus is emerging across service fields such as interaction design (Constantine & Lockwood, 2002), design (Evenson, 2008), that further research is needed to address these challenges. Some areas need particular attention, such as the growing complexity of service systems, the emergence of multichannel services, customer cocreation of service experiences, and the need for interdisciplinary methods. Together these trends have led to the emergence of service design as a new field (Mager, 2009) that takes a more holistic view of the service system. Service design is an emerging field (Mager, 2009) whose methods are still being developed and are often borrowed from related areas. Service design has been traditionally viewed as a specific stage of the new service development process (Edvardsson et al., 2000). However, the new service design field has adopted a broader approach, involving understanding users and their context, understanding service providers and social practices, and translating this understanding into the development of evidence and service systems interaction (Evenson, 2008). When designing complex systems, thinking with models helps bridge the gap between problem and solution: Models synthesize the understanding of users' needs and possible solutions in ways that help different stakeholders explore new ideas (Dubberly, Evenson & Robinson, 2008).

Service offerings today are enabled by complex service systems, which are configurations of people, technologies, and other resources that interact with other service systems to cocreate value (Maglio et al., 2009). Service systems can be modelled and designed at different levels. At the firm level, each of these service offerings is enabled by a firm's service system comprising multiple service interfaces such as physical stores, the telephone, or the Internet. At each service encounter, the customer interacts with a concrete service interface, which is a service subsystem that integrates the physical environment, people, and process performance. Different service system levels should be integrated into service design, but service design approaches have typically only focused on one system level at a time. Service design involves different components, such as the definition of the service concept, the service system, and the service process (Edvardsson et al., 2000). Although these components do not represent hierarchical levels, the service concept and the service system concept can be adapted for designing the different levels of service systems. Service innovation efforts have been hampered by their isolation in different academic disciplines and a lack of unifying models and languages. These problems have been major forces driving the emergence of IBM's Service Science, Management and Engineering (SSME) initiative, which encourages integrating work from different fields to develop the new

competencies required in the service-led economy (Chesbrough & Spohrer 2006; according to Partício et al., 2011). The creation of integrative methods, tools, and languages that unify these different perspectives is crucial for the development of the service design field. (Partício et al., 2011).

The term ‘service design’ was first introduced by Lynn Shostack (1984). The foundation of service design is in the 1990s and its roots are in a European design tradition (Wetter-Edman et al., 2014; Segelström & Holmlid, 2011). According to Wetter-Edman et al (2014, 109) during the last two decades “designers and design researchers have approached the service sector as a new potential partner for design, introducing a creative, human-centered, and iterative approach to service innovation (Sangiorgi 2009, Blomkvist et al. 2010, Pacenti and Sangiorgi 2010, Meroni and Sangiorgi 2011). Design-based approaches for service innovation include working with user centeredness, multidisciplinary teams, aesthetic and visual competence, and creative processes (Brown 2009, Kimbell 2009, Holmlid 2011).” Segelström and Holmlid (2011) argue that two conclusions point towards that service design is a design discipline rather than a service discipline, but their research also shows where there are gaps between the two which should be closed if the goal is to include service design in the service science family. (Segelström & Holmlid, 2011). Stickdorn et al. (2018) define service design as a multidisciplinary, but Segelström and Holmlid (2011) argue it is currently (at the time of the publication) more of a design discipline than a service discipline.

According to Stickdorn et al. (2018) service design can be observed from five perspectives: mindset, toolset, language, process, and management point of view (See figure 4). In this research the focus is on service design as a process but also toolset perspective is briefly examined.

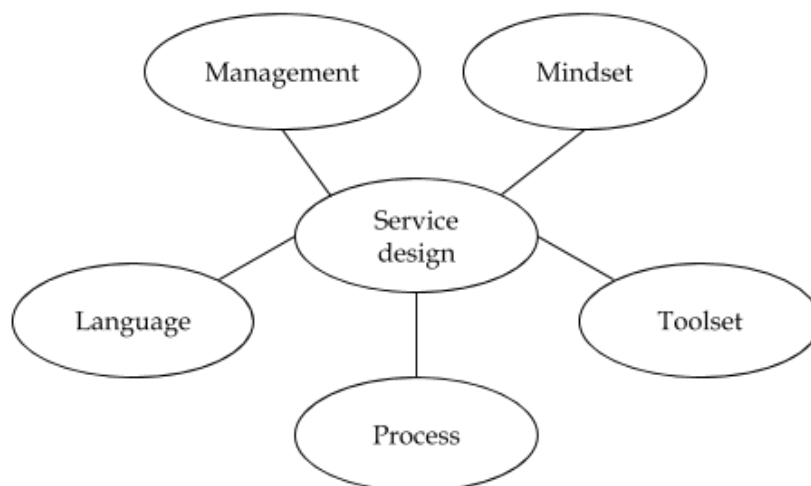


FIGURE 4 Five perspectives to service design (Stickdorn et al., 2018).

Service design is defined by Holmlid and Evenson (2008) as applying design methods and principles to the design of services. Service design is complimentary to conventional service development approaches and as such should become a contributor to services sciences for example (Holmlid

& Evenson, 2008). According to Wetter-Edman et al. (2014) service design is a practice that focuses on value in its experiential/heuristic/empirical dimension and proposes an outside-in approach to service innovation. The contextual experiences and human-centred design have been a much-discussed topic in service design practice for over two decades. The focus in service design is on observation and understanding users, facilitating collaboration and participation, “at the times and places where value is co-created” (Wetter-Edman et al., 2014, 107).

According to Stickdorn, Lawrence, Hormess and Schneider (2018, 27) service design is:

“... A practical approach to the creation and improvement of the offerings made by organizations. It has much in common with several other approaches like design thinking, experience design, and user experience design, has its origins in the design studio, and harmonizes well with service-dominant logic. It is a human-centred, collaborative, interdisciplinary, iterative approach which uses research, prototyping, and a set of easily understood activities and visualization tools to create and orchestrate experiences that meet the needs of the business, the user, and other stakeholders.” (Stickdorn et al., 2018, 27).

According to Mager (2008, 355) :

“Service designers visualise, formulate, and choreograph solutions to problems that do not necessarily exist today; they observe and interpret requirements and behavioural patterns and transform them into possible future services. This process applies explorative, generative, and evaluative design approaches, and the restructuring of existing services is as much a challenge in service design as the development of innovative new services.” (Mager, 2008, 355).

Some references (Mager, 2004; Hollins & Hollins, 1991; according to Sangiorgi, 2009) define service design as applying design methodologies into immaterial products and seeing services as products. Sangiorgi (2009, 416) notes that “the perspective that looks at services from the interaction point of view, is different from the one that was trying to define services as ‘products’ (Mager, 2004; Hollins & Hollins, 1991) and therefore as objects of a design process.” Hollins and Hollins (1991) and Mager (2004) suggest that services should be designed with the same attention to ‘products’, place the focus on the process (design management), with less emphasis on the specificity of services and therefore of design contribution (Sangiorgi, 2009). Yet, according to Stickdorn et al. (2018) service design approach comprehends everything as a service: the material products and immaterial products. Material product in the context of application development would be an application, the end-product of the development process. Immaterial product in the same context is designing the way the material product, application, is used or in development process: designing how the application development process goes. In this research the focus is in designing immaterial products and more specifically evaluating and

analysing the development process from service design point of view and designing the application development process. (Stickdorn et al., 2018).

Stickdorn et al (2018) mention in their book a broad category of terminology which people consider might use as a synonym for service design and the terms have many commonalities between each other and they are very similar. The differences are fewer than the common factors they share. These terms are such as holistic user experience (UX), experience design, design thinking, user-centred design, human-centred design, new marketing, and “even more” as Stickdorn et al (2018, 20) state. In this research the focus is on service design literature and therefore the differences between service design to other similar terms is only briefly discussed. The focus is in defining service design and exploring design-led development model, and not focusing on addressing differences or similarities of service design to other similar or close terms and defining those terms in detail.

TABLE 2 Principles of service design doing (Stickdorn et al., 2018, p. 27).

Principle	Description
Human-centred	Consider the experience of all the people affected by the service.
Collaborative	Stakeholders of various backgrounds and functions should be actively engaged in the service design process.
Iterative	Service design is an exploratory, adaptive, and experimental approach, iterating toward implementation.
Sequential	The service should be visualized and orchestrated as a sequence of interrelated actions.
Real	Needs should be researched in reality, ideas prototyped in reality, and intangible values evidenced as physical or digital reality.
Holistic	Services should sustainably address the needs of all stakeholders through the entire service and across the business.

The principle iterative means here starting with small and inexpensive attempts (such as light sketches and quick prototypes) and experiments, allowing them to fail, learning from the failure, and adapting the process along the way (Stickdorn et al., 2018).

3.1.1 Service design process

Service design process consists of iterative phases such as research, ideation, prototyping, and implementation (Stickdorn et al., 2018). There are multiple different methods that can be applied in the different phases of the process. As stated before, service design is a practical and cross-disciplinary approach which exploits a comprehensive toolset in the process.

According to Stickdorn et al. (2018) service design being "real" is one of the key principles and it refers to service design being fact-based. Therefore, the problem and the needs ought to be researched in reality. Some of the service design methods are utilized based on research and because of this, research is an important element and tool of service design. Nevertheless, the information

expressed using service design methods can be based on assumptions or research. Understanding and recognizing whether the content of tools use is based on research or assumptions helps to understand how credible a piece of work is and indicates how much you can challenge it. Often assumption-based tools develop into research-based tools over time as assumptions are challenged and research gaps are identified and closed through iterative research loops. The difference between assumption-based and research-based tools is following: In research-based tools, the content is based on research data. Pinpointing the research behind the content increases the credibility. Usually the research statement contains all the important aspects of the underlying research design, as an example describing the research methods used and the number of conducted interviews or observations. If the research is conducted properly, research-based tools have a greater significance than assumption-based tools. Research acquires more time and resources but in the end the findings are more robust and closer to reality. Assumption-based tools on the other hand are not based on research, but assumptions and the quality of the information depends on the teams/tool's creator's knowledge of the subject. Assumption-based tools can be distinguished to tools that are made 'ad hoc', a quick first draft used for planning and tools that have been created during a co-creative workshop. Depending if the people who participated in the workshop have profound knowledge of the subject, the quality of the tool created in the workshop can be quite good.

What sets service design apart from other disciplines, is the use of visualisation techniques in depicting the service being (re-)designed (Segelström & Holmlid, 2011). Visualisation has been stated as one of the most prominent features of service design by for example Segelström (2010), Kimbell (2009) and Mager (2008) (according to Segelström & Holmlid, 2011) and it has also been seen as one of the key distinguishing feature of service design by Segelström (2012). Segelström (2012, 175) found that "the various visualization tools all serve the purpose of communicating user data to different recipients. Additionally, the results points towards that there is a set of basic techniques, such as customer journeys and personas, which are almost universally used, as well as a long tail of techniques only used by a few companies. Finally, it is found that service designers to a large extent let the nature of the user data decide the form of visualisation together with the intended audience of the visualization." Visualisations are used by designers during the service development process as representations of existing or future services (Segelström & Holmlid, 2010). Holmlid (2007) draws the conclusion that service design is a highly visual design discipline. In this paragraph methods journey map (customer journey, touchpoints) and prototyping are presented. Next described is the design-led development process model which follows the service design principles.

3.1.2 Design-led development

Developing information systems, like enterprise applications, has many aspects and the user interface design (UI) is one of them. Design-led development is an agile development model where the main driver for development is the design of the application. According to Cooper (Fawcett, 2002), UI design should be first prototyped and tested by end users iteratively until validating and confirming the design. Prototypes can be quick, lightweight sketches at first and then improved (iteratively) while gaining understanding. After agreeing on the UI design, development can be started separately. This process is called a design-led development process (DLD). For prototyping there is also rapid application development (RAD) model which is an approach to software development that sees software being developed through a series of iterative stages generally involving the use of prototypes and high, active user involvement so as to speed up the development process (Cadle & Yates, 2009). Design-led development is explored in this research through SAP Build tool and its development process and functionalities which apply service design methods. Design-led development puts the user first by trying to understand their needs before implementing a solution. This requires investing time in user research and design at the beginning of the product lifecycle to ensure the creation of a software solution users will want to use.

Design-led development approach is explored by using SAP Build tool's model for SAP Fiori development as an example (See 4.1 and 4.3). As SAP Fiori application's one of the main elements is the user experience, in this section SAP Build, SAP's add-on product for SAP Fiori application development, is introduced and explained. The tool is explored in this research because it is based on service design methodology and follows a design-led development model. One of the research goals is to examine if using a tool like this would bring value to the SAP Fiori development, development process improvement, and the stakeholders involved in the process. SAP Build is a product provided by SAP for developing SAP Fiori applications. SAP Build implies SAP Fiori guidelines and elements to secure cohesiveness in the design and it follows design-led development process (definition in 3.2.1). The tool is a no-code platform for creating interactive UX prototypes for SAP Fiori applications. SAP Build can be used for free by free trial version or by a licenced version. With free trial version there are not all functionalities available. (SAP, 2020e.).

The application development process with SAP Build follows these three phases: Discover, Design and Develop. Under each phase are different steps to take. Discover phase is for exploring opportunities to add value and delight and to identify problem to solve and for whom after which talk to end users and uncover their real needs. Design phase is for creating the best solution based on the understanding of the problem and the needs. The best solution is created by prototyping, testing and iterating the design with end users. After testing the design with users, it is time to build the solution and to validate, adjust and release the product. (SAP, 2020e.).

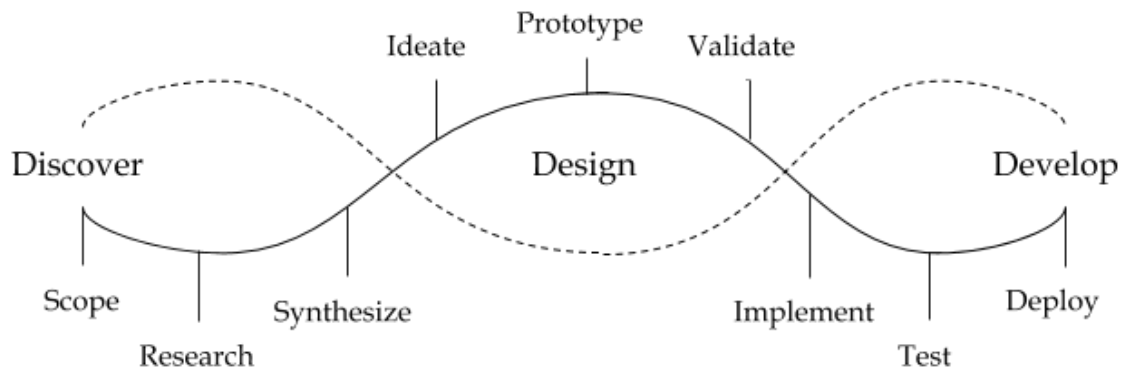


FIGURE 5 SAP Build application development process is design-led, and the process is based on service design methodology (SAP, 2020e).

In more detail, under the first phase, “Discover”, the required steps are scope, research and synthesize. In scope step the goal is to communicate a clear understanding of the problem, involve the right people and plan to maximize both their work and project time. This is done by creating a document and outlining the project commitments: What problems you are solving? What are your project deliverables and milestones? The goal in the next step, research, is to learn about users’ real needs and uncover opportunities to improve their life with your product. Research is done by interviewing the users which will provide the insights that will benefit from while building a solution. After scoping and researching comes synthesizing: In the synthesize step, the goal is to focus on the right problems, understand your users better and create personas for target users with detailed points of views. This is done by distilling findings, reframing the problem statement, creating a persona and defining points of views. Distilling findings is done by organizing the findings, identifying common needs and insights from research within the team to better understand your target user. Reframing the problem statement is done in order to create a common understanding of the goal you are addressing. By summarizing common user characteristics, needs and pain points into a single persona (Create a persona) to empathize with the target user and stay focused. By choosing to address a specific need to bring focus to the design, it will work as the anchor when creating the solution. (SAP Build, 2020e)

The mentioned methods for understanding the problem and needs are according to SAP Build tool, but there are also other service design methods that can be used outside of SAP Build: In designing experience-centric services, the customer journey and touchpoints are two important design process concepts (Zomerdijk & Voss 2009). Touchpoints occur whenever a customer interacts with the service provider across multiple channels and, therefore, are similar to service encounters (Bitner, Ostrom, & Meuter, 2000). The customer journey refers to a series of touchpoints, involving all activities and events related to the delivery of the service from the customer’s perspective. This view helps in understanding the service experience across multiple contacts but does

not offer an overall view of the service system structure or an integrated approach to the different levels of service design. (Patrício et al., 2011).

In the second phase, Design, the first action point is to “Ideate”. Ideating step is followed by prototyping and the design phase ends with validation step (Validate). The goal in the ideation step is to move from problem to finding a solution, by converging on the most desirable ideas and choose the best, and by assessing the idea’s viability and feasibility and visualize the user experience through storyboarding. This is done by creating use cases, brainstorming with the team and creating a storyboard and preparing a creative workspace. Use cases are created to understand how the personas use the system to accomplish the goal, how the system satisfies the goal, and what happens when issues occur. Brainstorming with the team is done in order to gather more ideas and more chances for a delightful solution. Brainstorming is done by generating a large variety of ideas quickly by bringing together a diverse set of minds. These steps are followed by storyboard creation which will help to combine the product ideas into a solution that will meet the user goals in a useful and delightful manner. Also, a part of this design phase is to prepare a creative workspace which will foster collaboration, creativity and innovation during of your design thinking project. (SAP, 2020e.).

The second step of design phase is to prototype. Prototyping is a fundamental element in service design. With prototyping the goal is to provide a low-fidelity version of your design, show the layout, interface and content, demonstrate actions and behaviours and validate early the solution with users. The starting point to achieve these goals is to create and share a low-fidelity prototype based on DT artefacts to make the best out of your DT workshop. Second step is to sketch a prototype by fleshing out the ideas quickly in a hand-sketched prototype and to create first concepts of possible solutions. This quickly hand-sketched prototype is shown to the users to get immediate feedback. Third part of this prototyping step is to create wireframes to visually represent the user interface design and foster a shared understanding of the design among the team and other stakeholders. Prototyping step leads to last action point of design phase which is validation. The aim in validate is to verify the focus of the research and synthesis, evaluate the effectiveness of the design and observe and listen to the users. This is done by testing the sketches with users by testing the designs, getting feedback and giving yourself a chance to make improvements before investing the time to develop a solution. (SAP, 2020e.).

The last phase, Develop, consists of implement, test and deploy actions where in implementation the goal is to bring the actual software solution to life. Implementation action covers also front-end to back-end development and helpful documentation which is done by User Story Mapping (USM) method. USM is an agile method that helps to get a better understanding of the big picture of the product and a common understanding of the user priorities. In test action the aim is to make sure the design works as planned, and meets performance goals, and to validate the desirability and viability of the product.

Test action is done by conducting a usability test which is important because this reveals early in the design or development phase how users interact with the design and to modify it if it is needed. The earlier these aspects are found out the better, as making changes becomes more difficult the further the development is. The goal of last action, deploy, is three folded: first to release your solution for final production, second is to deliver completed product to customers, and finally engage early adopters and begin gathering feedback by importing a completed and published prototype from SAP Build into SAP Web IDE. This will help to convert the prototype into a live application. (SAP, 2020e.).

3.2 Conclusions of the theory

This paragraph concludes the literature review. The purpose of this paragraph is to compare the plan-driven, agile and design-led application development approaches with service design process, and to conclude the literature review. Comparing any methodology objectively is difficult and the results is often based on the practitioner's subjective experiences and the author's intuitions (Song & Osterweil, 1991). The intent here is not to value one approach over another, although comparison often implies this. To overcome the limitations of comparison, a quasiformal comparison approach is applied: There are five different ways to approach quasiformal comparison, and in this research the fourth approach type is implied - Defining a metalanguage as a communication vehicle and a frame of reference against which methods are described (Sol, 1983). This was operationalized in the previous chapters by describing each method using the same typology: Introduction and definition, process characteristics, the challenges and redeeming features of the approach, with the exception of describing design-led development in more detail, as the subject is not as well known (Holmlid, 2007) as the two other development approaches presented in this research.

Service design strengths are in visualization and strive for a fact-based design, and for doing and documenting discussions visually from the beginning. Well aware of the challenges is plan-driven model, Royce (1987) has described additional steps for his model, to avoid the issues: In these steps, the need for documenting the design is mentioned, and it is recommended to repeat the step twice. The use of prototyping is common in agile approaches, but prototyping itself does not imply that the development process is agile: Prototyping is a fundamental method in design-led development, and prototyping techniques may be used in other types of development approaches as well (Cadle & Yates, 2009). Prototyping could be used in plan-driven approaches as well in the beginning and in the design phase to visualize the wanted solution.

TABLE 3 Comparison of development approaches based on literature.

	Plan-driven	Agile	Design-led
Process	Sequential; plan-driven	Iterative; collaborative	Iterative; collaborative; human-centred; fact-based; sequential; holistic
Size	Larger teams, projects and products	Smaller teams, projects and products	Project and product size can vary from small to large. Teams usually smaller but can vary
Requirements	Knowable early; largely stable	Largely emergent; rapid change	Rapid change
Design	At the beginning; Designed for current and foreseeable requirements	Throughout; Designed for current requirements	Throughout; Designed for current requirements
Prototyping	Can be used	Yes	Yes
End result	Known	Not known	Not known
Redesign	Expensive;	Inexpensive	Inexpensive
Dedicated resources required	No	Yes	Yes
Collaboration	Less frequently	Frequently; daily	Frequently

In this study the focus is on enterprise applications and enterprise application development process. Plan-driven and agile development processes were examined. The second part of literature review examined service design and design-led development process, where also challenges, pros and benefits of service design process were discussed. Few fundamental methods usually utilized during the process were presented. For each process approach examined, the challenges and redeeming features were discussed. After screening these aspects, a comparison of service design and application development process approaches was conducted by analysing the similarities and differences between these processes. As service design is a comprehensive/holistic approach it can be utilized in various cases, also when developing applications. Service design is not only fitted for one purpose but moreover it can be seen as a mindset, language, approach, and a process for any development cases. This chapter answered to what is service design and application development, what are the similarities and challenges in service design and application development processes.

Service design was defined as a practical approach to the creation and improvement of the offerings made by organizations which is a human-centred, collaborative, interdisciplinary, iterative approach which uses research, prototyping, and a set of easily understood activities and visualization tools, such as personas and journey maps, to create and orchestrate experiences that meet the needs of the business, the user, and other stakeholders. Service design has many similarities with several other approaches such as design thinking, experience design, and user experience design. Service design has its origins in the design studio and harmonizes well with service-dominant logic. Service

design can be explored from five different areas: a mindset, toolset, language, process and management approach. (Stickdorn et al., 2018.).

4 CASE: SAP FIORI APPLICATION DEVELOPMENT

The case company under examination is a global company providing lifecycle solutions to the marine and energy sector. The SAP ERP landscape is considered to be complex in the case company (Schutte, 2019). This chapter consists of defining enterprise resource planning system (ERP), SAP and SAP Business Suite 4 SAP HANA (SAP S/4HANA for short) and SAP Fiori. The SAP Fiori application development process today in the case company is examined, and lastly SAP Build tool and its design-led development model is introduced.

4.1 Enterprise Resource Planning

According to Esteves and Pastor (2001) enterprise resource planning (ERP) systems are software packages composed of several modules such as human resources, sales, finance and production, providing cross-organization integration of data through embedded business processes which can be customized for the needs of an organisation. Also, according to Jacobs and Weston (2007) the full ERP system is for serving all business processes within these the above-mentioned functional areas.

There are many ERP vendors, and SAP is one of the most known and largest ERP software and product providers globally. The chosen ERP system vendor for this research is SAP (Systemanalyse und Programmentwicklung) because it is the core ERP of the case company. The history of SAP (see figure 6) started in the early 1970s in 1972, to be exact, according to Jacobs and Weston (2007) and the company's purpose was to produce and market standard software for integrated business solutions (Jacobs & Weston, 2007).

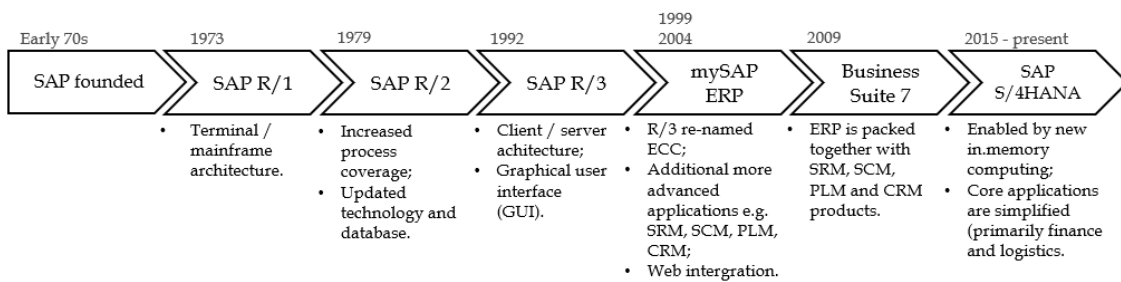


FIGURE 6 History of SAP (SAP, 2020b).

The current ERP system in the case company is SAP ERP (formerly named as SAP R/3) which supports all core business processes and functions required by today’s enterprises (Böder & Gröne, 2013). SAP ERP architecture consists of many modules (see figure 7). Architecture consists of fundamental concepts or properties of a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution (ISO/IEC/IEEE 24765:2017). The main user interface in SAP ERP is SAP GUI (graphical user interface).

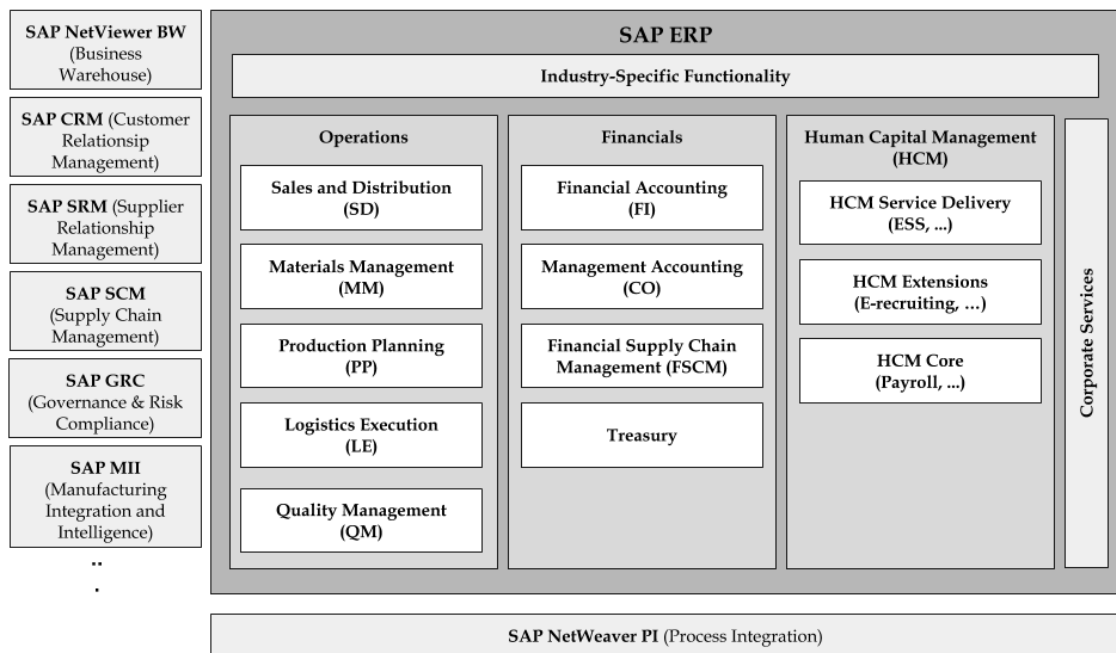


FIGURE 7 Overview of SAP ERP architecture (Böder & Gröne, 2013).

SAP ERP system is being replaced by SAP Business Suite 4 SAP HANA (SAP S/4HANA). According to SAP (2020a), SAP S/4HANA is an intelligent, integrated ERP system that runs on SAP’s in-memory database. It is a modern ERP that utilizes artificial intelligence (AI). SAP S/4HANA system can help to:

- Address industry-specific requirements with proven best practices for 26 verticals and enable new business models as your marketplace evolves
- Revolutionize business processes with intelligent automation – supported by artificial intelligence and robotic process automation
- Make better decisions faster with embedded analytics, a conversational interface, and digital assistants
- Meet your IT landscape goals with hybrid, cloud, and on-premise scenarios that share a consistent data model, code line, and user experience. (SAP, 2020a.).

In their research of ERP history, Jacobs and Weston (2007), presented their visions for the future, which were that ERP systems should enter an era of easy configuration. They describe that major corporations have realised the benefits in short implementation cycles, but also remind of the issues related to project management in large- and medium scale implementations which will still be significant in the future (Jacobs & Weston, 2007).

4.2 SAP Fiori

Moving towards SAP S/4HANA the role of SAP Fiori applications is ever greater as SAP Fiori is the overall user interface (SAP, 2020d). SAP Fiori architecture in SAP S/4HANA can be seen in figure 8 (SAP Help portal, 2020). According to SAP (2020d), SAP Fiori practices design-led development process which was presented in chapter 3.1.2. Design-led development. Following fundamental values are presented for SAP Fiori design: Consistency, integration, and intelligence, where consistency offers design solutions which can be adopted across SAP portfolio (SAP, 2020d). Integration means providing solutions for integrating different products into a coherent and easy to use environment (SAP, 2020d). Intelligence as the third value means establishing artificial intelligence and machine learning as an essential part of the user experience (SAP, 2020d).

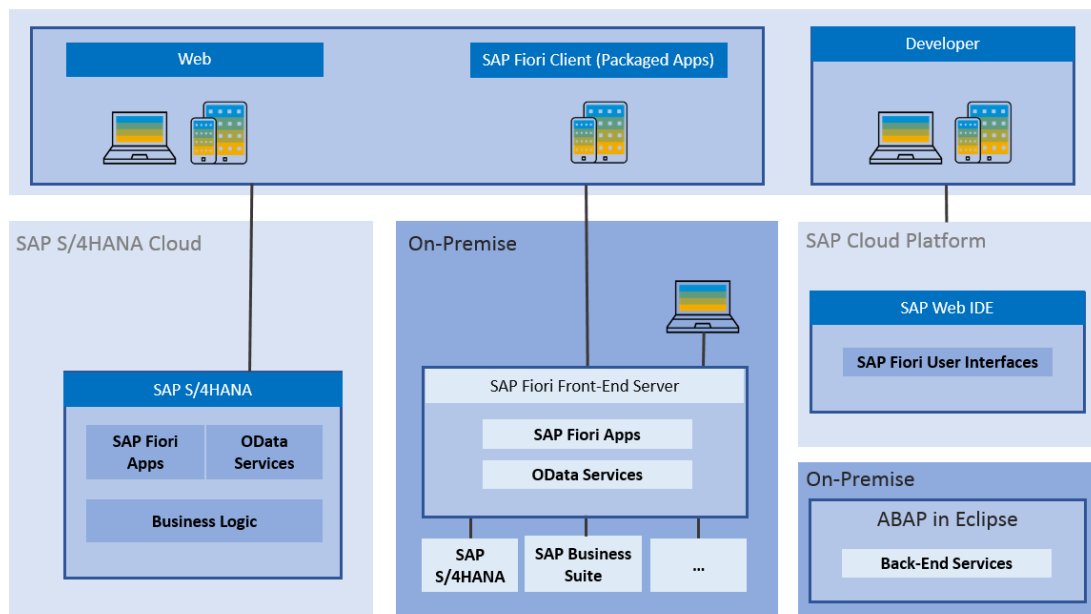


FIGURE 8 SAP Fiori architecture in SAP S/4HANA (SAP Help Portal, 2020).

In contrary to SAP GUI, SAP Fiori applications can be used in any platform or device (see figure 9) which enables new ways of working as users can work where they want and on the device of their choice (SAP, 2020d).

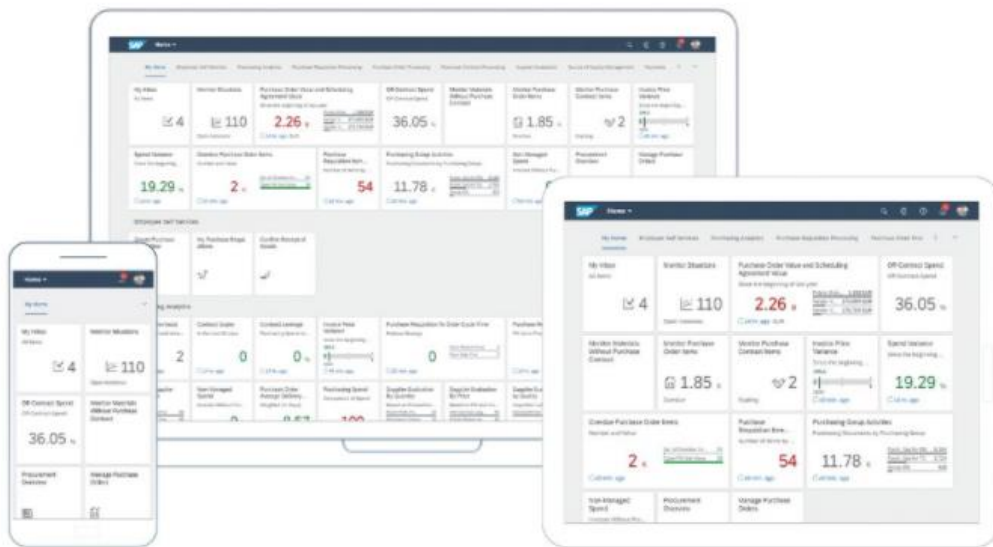


FIGURE 9 SAP Fiori applications can be used in any platform (SAP, 2020c).

The interface is a collection of Fiori applications that are built according to specific SAP Fiori guidelines and design principles (See Figure 10). Fiori applications are based on five guidelines. Firstly, the applications are role-based so the applications are designed for user's role, based on their needs and how they work. Applications are adaptive as they adapt to multiple use cases and

devises (See figure 9). The user interface is coherent and simple, because it provides one fluid and intuitive experience and includes only what is necessary. Last guideline is that it is delightful and makes an emotional connection with the user and enriches user's work experience. (SAP, 2020d.).

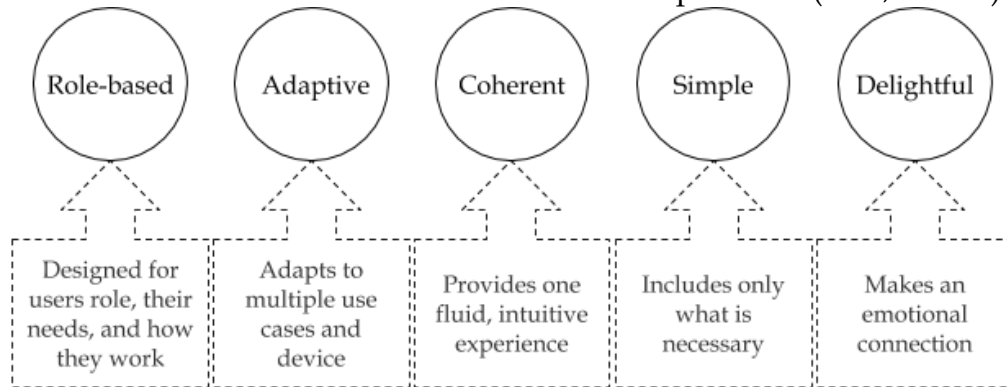


FIGURE 10 SAP Fiori design principles (SAP, 2020d).

According to Szirtes & Rivlin (2015) the best SAP Fiori option is to use a standard, out of the box SAP Fiori application. In case standard application is inadequate, a suitable standard application can be extended if there is a functional gap. If no suitable standard SAP Fiori application is available, it is possible to create a custom SAP Fiori application. (Szirtes & Rivlin, 2015). SAP Fiori benefits for business are both human and monetary (see figures 11 and 12) (SAP, 2020d), as the applications gain more productivity and data quality, decrease change requests and user errors, and save on training costs.

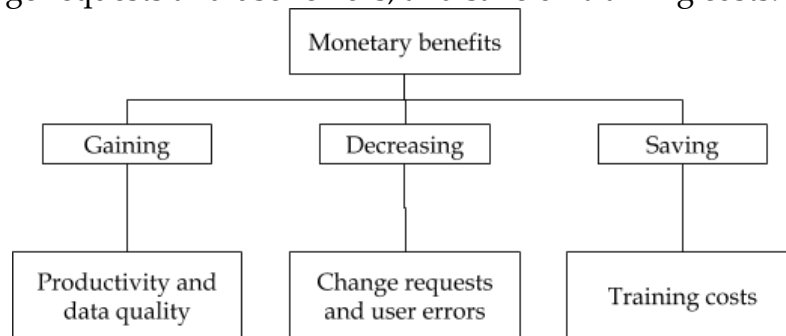


FIGURE 11 SAP Fiori monetary benefits for business (SAP, 2020d).

For the human benefits (see figure 11), SAP Fiori is also promised to increase customer loyalty, solution adaptation and user satisfaction, and strengthen the relationship between IT and business (SAP, 2020d).

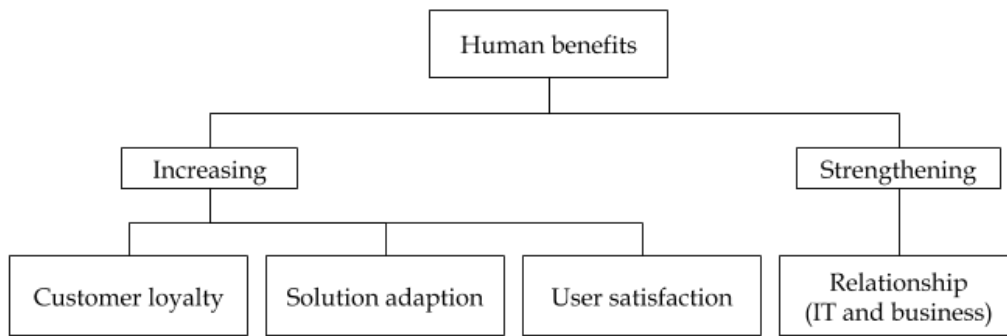


FIGURE 12 SAP Fiori human benefits for business (SAP, 2020d).

The difference between user experience and user interface (See figure 13) according to SAP is that user experience focuses on understanding end user's needs, the engagement and usability of the solution whereas user interface is more focused on the interface's layout, visual design, and branding (SAP, 2020d). SAP Fiori brings personalization possibilities for the user to the user interface which is a way to tailor the user experience. Personalization means that there is a default design provided but the personalization can be used to tailor the user experience to meet individual user's needs and preferences (World Wide Web Consortium, 2020). SAP Fiori user interface is a major leap from the transactional GUI to this century as it takes the user experience to another level. According to Jones and Marsden (2006) good user experience is a sum of several different aspects, such as the product such as application should respond to user needs, provide a solid and easy-to-learn user interface, and offer pleasant experiences in general. There are different concepts closely similar to each other such as customer-centred, user-centred and human-centred design. In this research we use the term human-centred because it can involve any stakeholder without excluding another one. In human-centred design the focus is on making solutions usable and pleasurable for the humans involved in achieving the solutions (Wetter-Edman, et al., 2014). The term "human" is used in favour of the more commonly encountered "user," because the "human-centred" approach considers the importance and role of a larger network of actors, not only users, who are directly or indirectly involved in the service provision and use (Rizzo, 2010; Meroni & Sangiorgi 2011).

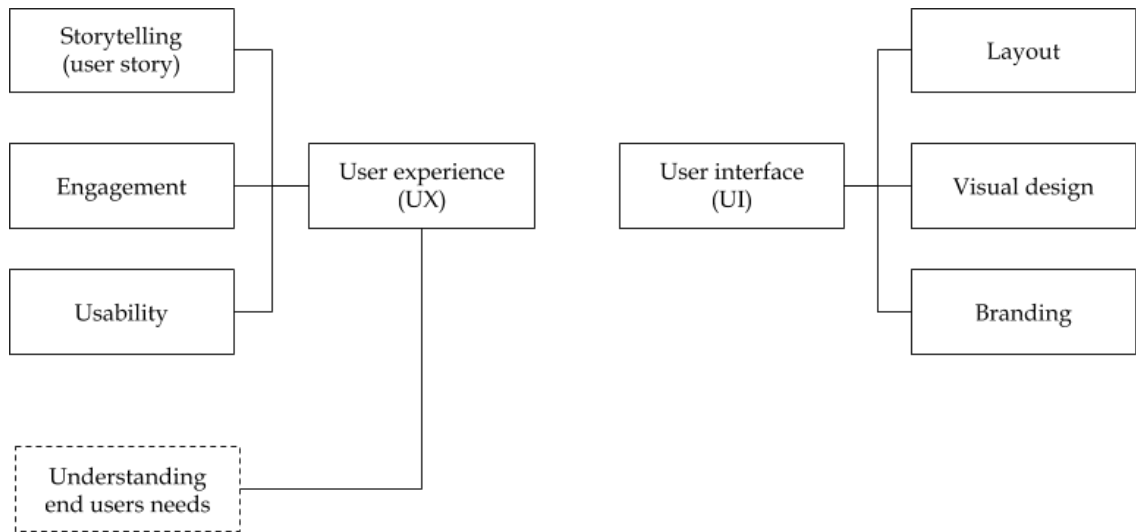


FIGURE 13 Differences between user experience and user interface according to SAP (2020d).

In this paragraph SAP Fiori was introduced and its architecture, guidelines and benefits. SAP Fiori applications can be used in any platform and if standard application is not suitable, there are extensibility as well as customisation possibilities available. SAP Fiori user interface differs from the transactional SAP GUI as the applications are role-based and can be personalized for the user's needs to some extent. Where SAP GUI can be complex for a new user to learn and more training is needed, SAP Fiori applications are more modern and brought to this century, where the user experience is better, and training is easier.

4.3 SAP development process today

Development strategy is organization's overall plan of development, describing the effective use of resources in support of the organization in its future activities which involves setting objectives and proposing initiatives for action (ISO/IEC/IEEE 24765:2017). In this part the SAP Fiori application development process in case company is presented. New SAP Fiori developments should follow the same development process as any new SAP development. Case company's current development model is a customized version of the plan-driven waterfall model (see 2.2.1 Plan-driven development). The traditional waterfall model is a structured development method, where testing is performed after the development is completed, and it contains minimal customer involvement. (Schutte, 2019). Enhancement request (ER) is used in the case company to develop or enhance the system or functionalities.

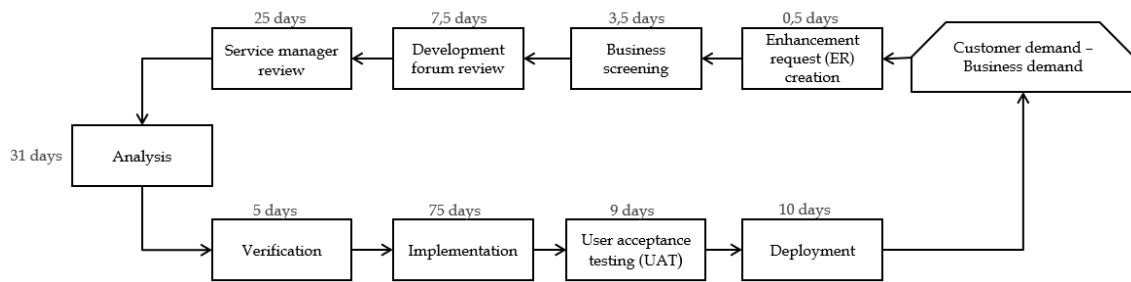


FIGURE 14 Overall enhancement request process for all SAP development and average duration of each step in the case company (Schutte, 2019).

The constraints of SAP software development in case organization have been collected by Schutte (2019) via desk research, data analysis, survey and interviews. The results are sevenfold:

1. One process for a wide variety of different software change requests;
2. A high amount of bureaucratic procedures and supporting systems;
3. High work in progress for the development streams, which leads to long lead times;
4. Lacking control on the work in progress and missing visibility for the involved stakeholders;
5. Unclear added value for software change requests and missing cross functional alignment;
6. Missing learning opportunities for the involved stakeholders due to a high varying involvement;
7. The process not supporting the desired collaboration in between the business, IM and developers very well.

The stakeholders in SAP software development process in case organization are following: Business operation, business development, information management (IM) organization and system developers (Figure 15). Stakeholders are individuals or groups who are affected by the activities of an organization or a project (Cadle & Yates, 2009). Explain all stakeholders' tasks involved in the process. In this research the focus stakeholder is business development (business for short from now on) and as a minor stakeholder IM organization is also concerned with. System developers are referred to as externals in this study. The research focus is in business stakeholders and as described in data collection business interviewees were asked about collaboration with other stakeholders; IM and externals (See chapter 5).

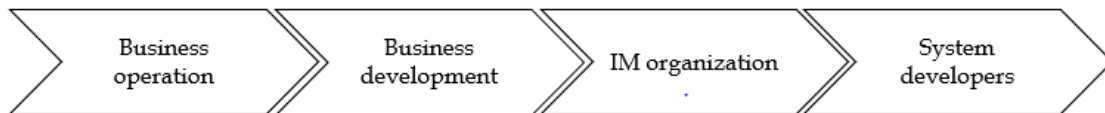


FIGURE 15 Stakeholders in SAP development (Schutte, 2019).

There are collaboration issues regarding SAP developments within the company such as missing active collaboration and understanding between business, IM and developers as well as no real accountability for external developer performance. Collaboration challenges are retrieved according to Schutte (2019) from following reasons:

- Globally spread stakeholders;
- Current contractual setup with external vendors;
- Demand from financial process owners for fixed price development;
- No clear product definitions;
- Since end of 2018 high focus on cost savings and not allowing external consultants. (Schutte, 2019).

Collaboration is one of the seven main challenges and it is analysed whether service design approach would have some possibilities to answer this.

4.4 Summary

In this chapter ERP, SAP and SAP S/4HANA and SAP Fiori were presented. The case company was shortly introduced, and its overall SAP development process today for all SAP developments was examined. Some challenges and key aspects regarding case company's SAP developments were introduced according to case research conducted by Schutte (2019) to gain more understanding of the general development process and challenges faced today. However, with SAP Fiori development there might be potential to reduce development time with the help of SAP Build tool (presented in 3.1.2) to create fast mock-up solutions. According to SAP (2020d) SAP Fiori benefits different stakeholders by inter alia gaining productivity and decreasing change requests and user errors, strengthens the relationship between IT and business and increases user satisfaction.

5 RESEARCH METHODS AND ANALYSIS

This study applies qualitative research methods. The process of choosing the research methods, data collection and data analysis are presented and defined in this chapter in the mentioned order.

5.1 Choosing the methods

This research is a qualitative case study. This method was chosen by comparing quantitative and qualitative research methods to each other by estimating their suitability to execute the study. There are few different case study types, such as exploratory, descriptive, and explanatory. This case study is instrumental case study as the case company's SAP Fiori application development is studied to understand SAP Fiori application development as a larger concept (Stake, 1995). Qualitative research's goal is to understand research object's quality, characteristics and importance (Hirsjärvi, Remes & Sajavaara, 2009, 161). Research object and phenomenon of this study is a development process of SAP Fiori applications in the case company. Specially experiences of development process are under observation to gain understanding of what are the challenges today and hopes for future. A better understanding of the research phenomena is pursued from business point of view with empirical research methods by going through the whole SAP Fiori development project from noticing a need to deployment. Case study is one of qualitative methods (Benbasat, Goldstein & Mead, 1987) and it is the chosen research strategy because the intention is to collect detailed, intensive information of a single case or small group of related cases (Hirsjärvi et al., 2009, 134). As a research strategy case study is most compatible in describing individuals, groups or community's cases where the focus of interest is often in processes. Theory or testing hypotheses is not the starting point with case study but detailed and complex examination and analysis of collected data (Hirsjärvi et al., 2009, 164).

A case study examines a phenomenon in its natural setting, employing multiple methods of data collection to gather information from one or a few entities (people, groups, or organizations). The boundaries of the phenomenon are not clearly evident at the outset of the research and no experimental control or manipulation is used (Benbasat et al., 1987, 369).

The focus of case research and other qualitative research methods is on understanding social phenomena in their natural setting and cultural context (Myers, 1998, according to Darke et al., 1998). Case research according to Darke, Shanks & Broadbent (1998) is well suited to understanding the interactions between information technology-related innovations and organizational contexts. There are two approaches to case study research: The positivist and the interpretivist approaches. (Darke et al., 1998). There are three reasons why case study research is a viable information systems research strategy according to Benbasat et al. (1987, 370). First is that the researcher can study information systems in a natural setting, learn about the state of the art, and generate theories from practice. Secondly, the case method allows the researcher to answer "how" and "why" questions, that is, to understand the nature and complexity of the processes taking place. Questions such as, "How does a manager effectively introduce new information technologies?" are critical ones for researchers to pursue. A case approach is an appropriate way to research an area in which few previous studies have been carried out. With the rapid pace of change in the information systems field, many new topics emerge each year for which valuable insights can be gained through the use of case research." Case study is seen as particularly appropriate for certain types of problems: For cases where research and theory are at early stages, and for practice-based problems where the context of action is critical, and the experiences of the actors are important (Benbasat et al., 1987, 369).

In case study the data is collected by using multiple different methods such as observation, interviews and examining documents (Hirsjärvi et al., 2009, 135). The chosen research techniques consist of a literature review, for data collection open-ended interviews were conducted, and for data analysis qualitative content analysis was applied. Also two other qualitative research methods, observation and written notes, were planned to be a part of research methods but due to changes in the global environment caused by Covid-19 pandemic and its affect also to the economic situation during spring 2020, all SAP Fiori projects were put on hold until further notice in the case company and therefore all planned actions were not implemented. The qualitative interview is the most common and one of the most important data gathering tools in qualitative research (Myers & Newman, 2007, 3). The interview structure can be found in the appendix 1. For detailed examination of data, the interview recordings are transcript for easier data analysis (Hirsjärvi et al., 2009, 222).

The chosen qualitative research technique for data analysis is content analysis because the goal of content analysis is "to provide knowledge and understanding of the phenomenon under study" (Downe-Wamboldt, 1992,

314). There are three distinct approaches to content analysis: conventional, directed, and summative. These approaches are used to interpret meaning from the content of text data and adhere to the naturalistic paradigm. (Hsieh & Shannon, 2005). Text data to be analysed can be in various forms: Text data can be in verbal, print, or electronic form and it might have been obtained from narrative responses, open-ended survey questions, interviews, focus groups, observations, or print media such as articles, books, or manuals (Kondracki & Wellman, 2002; according to Hsieh & Shannon, 2005). Text data of this study is in electronic form and obtained from interviews and a survey questionnaire answers. A similar analytical process is required for all three qualitative content analysis approaches. The analytical approach consists of seven classis steps that are formulating the research questions to be answered, selecting the sample to be analysed, defining the categories to be applied, outlining the coding process and the coder training, implementing the coding process, determining trustworthiness, and analysing the results of the coding process (Kaid, 1989 ; according to Hsieh & Shannon, 2005). Deciding which content analysis approach needs to be used can be done by matching the specific research purpose and the state of science in the area of interest with the appropriate analysis technique (Hsieh & Shannon, 2005). The qualitative content analysis approach used in this research is conventional content analysis, because usually a richer understanding of a phenomenon of interest is gained with this approach. Also, another advantage of the chosen approach is that direct information can be gained from research interviewees without imposing preconceived categories or theoretical perspectives. In a conventional content analysis, categories are derived from text data during data analysis. (Hsieh & Shannon, 2005).

The success of a content analysis depends greatly on the coding process and to the trustworthiness of the research the development of the coding is central (Folger, Hewes, & Poole, 1984; according to Hsieh & Shannon 2005; Hsieh & Shannon, 2005). According to Weber (1990; according to Hsieh & Shannon, 2005) In content analysis the basic coding process is organizing large amounts of text into much fewer content categories which is followed by identifying the relationships among categories. A coding scheme is created in the coding process to guide researchers to make decisions in the content analysis and it includes the process and rules of data analysis that are scientific, logical, and systematic. (Hsieh & Shannon, 2005).

5.2 Data collection

In this chapter, the process of data collection is explained. Data was chosen to be collected by interviews which is a common qualitative research method. The interviewees were nominated by representative person from the case organization as they had knowledge of who would fit the research purpose. The representative person provided five businessperson's names and researcher

contacted them first via Skype to ask whether they would be interested in taking part. After granting a permission, the official interview invitation was sent to the person via email with a short description and motivation of the study and suggested interview time. The interview time was agreed by the interviewee. All five of the interviews were conducted by the author with a single interviewee via Microsoft Teams virtual meetings. The interviews were conducted between May 25th and May 27th, 2020. The interviewees were all located in Finland and career lengths in case company ranged from 11 to 28 years. All interviewees had experience of at least one development process of SAP Fiori projects, but experience also ranged from one SAP Fiori development to more (also, in two cases the project could involve developing two or more applications relating to same subject). Also, some of the interviewees had quite recent experience and for some the experience was from few years ago and this possibly had some effect as the memories were not as freshly in mind. Three out of five interviewees had experience of SAP Fiori development with SAP Build tool.

The overall focus of the interviews was to collect data about experiences of SAP Fiori application development projects in the case company from business point of view and hear their opinions and comments about design-led-development model and SAP Build tool. The open-ended interviews were semi-structured and focused on warm up questions, SAP Fiori development today, SAP Fiori development in the future, design led development with SAP Build, and interview was ended with a closing question. Notes were taken during the interviews and all interviews were recorded. The total interview duration was 4 h 7 min, with a median of 54 :55 minutes. The lengths of the interviews in minutes were: 39 :53, 56 :01, 31 :54, 60 :11min, and 54 :55 minutes.

5.3 Data analysis

The goal of this study is to learn more about the SAP Fiori development process experiences in the case company to address their needs and pain points in the process today and opportunities for the future state. Also, to hear their comments on design led development to understand if service design methodology would be good/possible fit to the case company for the future (or some aspects of it). Trying to understand if there is potential service design could be integrated to the SAP Fiori development process and if so, how and what possible solutions service design has to offer for the challenges faced today. Because there is no existing theory, the research questions are:

1. What kind of experiences of SAP Fiori development process and SAP Build tool business has and what kind of hopes there are for future development?
2. What kind of methods are used in business side for SAP Fiori application development?

3. How IM's role is perceived by business in the development process and what kind of amendments and aspirations business has for future?

The meaning of the study is to produce comprehensive answers to research questions of SAP Fiori developments in real company and therefore qualitative methodology is seen most suited for the realisation of this research. Because of the purpose of the study, the target sample size is also meaningfully chosen instead of using a stratified sampling technique. (Hirsjärvi et al., 2009, 164). The target sample size was five businesspersons who had experience of one or more SAP Fiori developments in case company. Experience of SAP Build tool was not mandatory, but it was known prior to recruitment and data collection that at least one of them had some experience of the tool.

Prior to recruitment and data collection the research procedures were approved by case company representative. Informed consent was obtained from all interviewees. Data was collected through individual interviews using open-ended questions. All interviews were recorded and transcribed verbatim. Interviews were held individually via Microsoft Teams platform. Also group interviews would have been an option, but because of the virtual setting it would have been a challenge in some aspects such as possible technical problems, facilitation issues, not as easy to engage or to be activate a group of people, and scheduling the interview to suit all. Microsoft Teams platform was chosen as the interview meeting platform because it is the main platform used in the case company for virtual meetings and therefore interviewees had easy access to the platform and the platform was already familiar to the interviewed person. Microsoft Teams platform was chosen also because of the easy recording functionality. The main reason why interviews were held virtually was the state of global environment at the research time and the request by Finnish government to avoid physical contact due to COVID-19 pandemic. Interviewees were also located in different cities so due to travel restrictions and lack of research funding there would not have been the possibility to travel for the interviews.

Data is analysed according to conventional content data analysis (Hsieh & Shannon, 2005). The analysis started by reading each interview transcript from beginning to end, as one would read a novel. Then each transcript was read carefully, highlighting text that appeared to describe experiences, challenges, wishes and writing in the Excel document a keyword or phrase that seemed to capture the essence, by using the interviewee's words. Working through the transcript, attempting to limit these developing codes as much as possible. After open coding two of the transcripts, the preliminary codes were decided and then the remaining transcripts were coded by using chosen codes and adding new codes when encountered with data that did not fit to an existing code. Original transcripts were also recoded accordingly. Once all transcripts were coded, all data was examined within particular code. Some codes were combined during this process, whereas others were split into subcategories.

Finally, the final codes were examined and organized into hierarchical structure if possible.

With a conventional approach to content analysis, relevant theories or other research findings are addressed in the discussion section of the study to compare and contrast findings to theory. As according to conventional content analysis, the discussion includes a summary of how the findings from study contribute to knowledge in the area of interest and suggestions for practice, teaching, and future research. (Hsieh & Shannon, 2005, 1279).

6 RESULTS

The results from the interviews are presented in this chapter, in connection to the research questions which helped answer them. The interview structure can be found in appendix 1. First the results SAP Fiori development process with and without SAP Build tool are presented. This is followed by section includes results of the experiences of SAP Fiori developments: the challenges and positive factors in the process and in collaboration between different stakeholders. The role of IM is also discussed in this chapter. Then the used prototyping methods in the development process are presented after which experiences and thoughts of SAP Build tool and its usage and design-led development model are presented. Finally, thoughts and ideas for future development are shared.

As the case company moves to SAP S/4HANA there will be a lot of SAP Fiori application development, an interviewee (5) comments. They continue that “it is good to plan it [SAP Fiori development process] well now and how developments are handled in the company so that the infrastructure and organisation is ready and we can effectively develop those, instead of thinking about this when the requirements come.” (Interviewee 5). This describes quite well the motivation for the research as now is a good time to evaluate what is working and what is not, and what are the wishes towards the future, so that these issues can be taken into consideration early on. See table 4 for the interviewee identification information.

TABLE 4 Interviewee identification information.

Interviewee	Years in the company	SAP Fiori developments (number of applications)
Interviewee 1	13	2
Interviewee 2	28	2
Interviewee 3	15	4 (developed around the same idea)
Interviewee 4	16	4-5 (incl. projects where project owner or giving input to projects)
Interviewee 5	11	1 (as a part of bigger project; in addition, has given input to other Fiori developments)

6.1 SAP Fiori development process

Based on the interview data, an overall picture of the development process of SAP Fiori projects was able to be comprehended. The process began with recognizing a need or gap in the current solution or by acknowledging that this type of solution could be made. In one case there was not a decision initially that a Fiori application would be the solution for the need, but many different solutions were examined and then decided to do the Fiori application as one part of the project as the user interface for users. Another part of the project was related to the analytics and for it there was other vendor. In one Fiori project a standard SAP application was used as a reference or as a base for customized solution and first the standard applications were explored to understand what changes or additions were needed. In the beginning there have been discussions withing the company with different business organisations and in some cases also with IM. In one case approximately half a year went to the initial discussions to have common understanding what was needed from the solution. The requirement mapping has been done in a business-driven way. Prototyping has been used in the planning and design phase to visually present the needed solution for example to external vendors.

In development phase external vendors have been responsible for the development of the solution. External vendors have done their own testing and testing has been also done by business. According to the customized waterfall process that is used in the case company, testing includes user acceptance testing (UAT) phase. There can be many UAT rounds before moving to production phase. In one case there were three different UAT rounds before it was moved to production for the first version. In this case, after getting feedback from the UAT round, the solution was improved and changed according to feedback, following next round of testing. Final phase for the development process was production. After the production and ending the project, applications have been improved with enhancements, where the modifications have been smaller so there has not been as extensive development process, as the changes are enhancing the existing solution.

With SAP Build tool the development process was slightly different. All three SAP Fiori projects used the free-trial version of the tool. In the free trial there were fewer functionalities and for one it seemed better for proof of concept type of need and use. As one interviewee put it, "it would not have made sense to pay for the license for one development." In the same case as there was no readiness to pay for the license, the use of the tool was discontinued due to the shortcomings in the free trial version. The prototyping started by first creating quick prototypes or rapid prototypes by drawing the possible solution to paper. Then they started to design the clickable prototype with the tool. While creating the prototype the solution was discussed in the corresponsive business organisation internally and they also already did demo sessions to the users and gathered feedback from them. When they were happy

with the prototype and how the solution should work, they made more detailed specification and submitted requests of proposals from different vendors. In one of the three cases, prototyping with SAP Build for creating and designing the screenshots was done in collaboration with the external vendor. Together in a workshop held virtually or face-to-face, external vendor would use the tool to create and design prototypes while project owner would follow and comment how the prototypes should be. In two of the cases the tool was used by the project team: In the first case, one team member used the tool for designing the prototypes. This case was also done without creating an official development project of it. They did lot of the planning and ideating as well as creating the logic for the solution as much as they could before selecting the vendor and left the things they could not resolve to the vendor. In the second case the interviewee had used the tool and created the screens. For application enhancements SAP Build tool has not been used to create clickable prototypes because enhancements have been additions to the application. Instead at least in one case anyway pictures of user interface have been made where the location of the additional elements can be seen.

6.2 SAP Fiori development experiences

Compared to traditional SAP development, it was commonly agreed that SAP Fiori applications have better user experience which makes it more pleasing for the end users. Also, SAP Fiori was seen as a missing piece (Interview 4 and 5):

“In the SAP ERP the back-end side things are functioning but a user interface that would be easy and simple has been missing so Fiori brings now whole new possibilities.” (Interviewee 4).

SAP Fiori applications were generally found to be more easily learned by end users. The user interface of SAP Fiori applications was mentioned as the main difference to traditional SAP development. One interviewee contemplated that for new users SAP Fiori applications are easier to take into use as they follow more modern and familiar look and feel of a web user interfaces as oppose to SAP GUI. At the same time, another interviewee considered that at least for some users with experience in SAP GUI, a change in attitudes is required in order to make them implement the new user interface successfully. As one interviewee (5) said it, some of the users “love that GUI” and “it could be a big part of their professionalism and professional pride as they for example remember all the transactions by heart that no one else does as SAP is considered as a complex system,” and continued that maybe it does not feel great that someone then makes a simple and easy version of their knowledge area that anybody can use. As challenge or concern with SAP Fiori applications was the need for customization and whether SAP’s standard applications have any value in them out-of-the-box. Regarding this topic one interviewee

mentioned that with customization there are the endless possibilities with SAP Fiori applications but was also concerned that the standard applications might not bring any added value.

The results regarding the experiences of SAP Fiori development processes are addressed and described in this section, including the challenges and positive aspects during different phases of the development process, collaboration issues, and thoughts about the result. Overall, there are the same challenges as in traditional SAP development, one interviewee states, especially if you would like to build new complex logic. Developing Fiori application would be more straightforward if there would not be need for complex customisation. Many were suspicious whether the standard applications bring added value and whether standard applications can be taken into use as such.

At the beginning phase of the development there were challenges such as time consuming discussions internally and finding consensus that new application is needed and how the problem could be solved: Within business there are many different organisations and in addition also other organisations in the company, and all these different organisations can have different opinions on how something should be done. What was also mentioned as time consuming in the beginning was identifying the problem, as the problem should be known quite well to know how it could be solved. One interviewee mentioned that there were challenges with requirements mapping at the beginning but it has fixed to some extent so there is not anymore same challenge as before. Experiences in the planning phase explicitly were that planning with the externals went smoothly and business should be involved in the designing of the layout: Involving business users added the approval for the application among users when they had the experience of contributing to the application and this had positive effects on engaging users to testing and then to the implementation of the new application.

In the development phase quality issues came up in two interviews: The quality of the code and the quality of data. The data quality should be good otherwise it does not necessarily serve the purpose: In order to have good user interface there should be a functioning process behind it and the process should use good quality data (Interview 4). The development of the solution is done by externals and regarding the quality of the code there was uncertainty by one interviewee whether the code was good quality (Interviewee 1). In this case external did not understand how the application will be used regardless of efforts. This was one of the firsts projects for the external and since then this issue has improved. The uncertainty regarding the code quality came up from the testing challenges in this case which are presented next.

Testing phase in the case mentioned above was problematic and laborious: Every time something was modified in the application, everything had to be tested as something would break always. Externals tested the solution as well, but they tested only the part which had been modified and not the entire application or different ways of using the application so it was up to business to test it when something broke which had no relation to the modified part of the

solution. Testing was very time consuming and it would a week of work from one person to test the whole solution. Regarding testing there were also positive aspects and ways of working that were found functioning. Although there were challenges in the testing it was organised: End users from all related businesses were involved and there was one common Excel-sheet which was used by everyone involved, where every comment, problem, or question were added. External development team added their comments to the same list. Based on the list, the decisions about priority and criticality were made. In other case the development process throughout was executed in a quite agile way and the iterative testing of two internal team members with externals all through the development was successful. It was an efficient way of finding and solving inconvenient functionalities that otherwise would have been missed, in addition to removing bugs.

After production some challenges and issues also arise such as effects of Fiori patching which messed up the application in one case. This was also related to the one case with testing challenges and quality of data issues. The interviewee is concerned whether this will happen with next Fiori patching as well (Interviewee 1). Another external developer did not know how to fix the tool, so they had to buy the repair work from the external who had developed the application. Therefore, organizing a tender competition was not possible in this case. One challenge was the end user's disappointment after the first version was implemented as they had given a lot of input and it was not that much included in the first version. With enhancements the application has been improved and the user experience is better, but it was a big disappointment in the beginning. Despite these issues, there were also positive comments that the newest application has worked surprisingly well and better than was expected, so there has been improvement after first application development experiences. After collaboration experiences the thoughts of the end results are addressed in more detail.

Although an interviewee (3) said it is difficult to give any general grade for the application and the first version of one application was a big disappointment (Interviewee 1), the other interviewee's thoughts about the final outcome of the application were mainly positive:

"Excellent outcome." (Interviewee 2).

"What has been done so far has been positive." (Interviewee 4).

"One of the few things that have been developed that has been so easily implemented." --- "It has worked well, and it enabled persons who were not SAP specialists to use it with very minimal SAP training". (Interviewee 5).

However there has been also challenges, as in one case some functionalities have not been as designed and discussed in the final application and then there has been discussions with the externals (Interviewee 3). Some sections work better than others in case, but they had succeeded in automating

a part the process with the application (Interviewee 3). In another case the first version was very disappointing for the users as it lacked functionalities which they had hoped for but after the third version the experience has been better apart from the layout and scalability issues (Interviewee 1). The layout and scalability issues are today the negative feedback that are received regarding this application. When asked about what kind of feedback the interviewees have received afterwards from the end users, the answer is that feedback has been mainly positive but also negative to some extent. Especially in the beginning of the implementation some users perceived the new application as more complex than the current one but after time feedback has been more positive. In some cases (interviewees 1 and 4), the application has been developed in multiple phases and is still being enhanced, therefore the application is not ready, but the feedback has been mainly positive in these cases too apart from above-mentioned negative feedbacks. In one case (5) some negative feedback has been coming from the users who would like to use SAP GUI, the old interface.

6.3 Collaboration experiences

Next the results of collaboration among stakeholders during the development process are presented. First presented are the collaboration challenges and success with externals, followed by business results and IM results. According to one interviewee the collaboration worked in every relation, and other interviewees also thought the collaboration is mainly positive while there are certain challenges and areas to for improvement. One challenge that is evident with all stakeholders is the resource allocation: For example, many works in multiple different projects. Because of lack of time for example business end users have their hands full with their daily work which makes engaging them and allocating time for development process for user input difficult. One interviewee wondered that in agile process this might be an issue if a resource is unable to give input for longer time and if it makes the project stand still until that.

There were both positive and negative comments when asked about collaboration with external stakeholders. The dependency of externals was mentioned by two interviewees as a concern and challenge: In one case they would have wanted to proceed faster but the external provider could not allocate resources according to the business wish. Also earlier mentioned quality issue was described as a challenge with external and related to that, the dependency on externals if something for example breaks: There is a concern that the development will begin to cost a lot more at some point, and because there is no internal knowhow, they wondered if they are too dependent on the externals. At this point it was described by one interviewee that it is important to find a good external developer, especially if there is no knowledge in-house. In one case the external activity was described as “eventful” as in the beginning

some key person had left the external company and in the end the vendor had to get the Fiori development from outside. In the end however, the developer who joined the project was very good according to the interviewee and they had good collaboration to get the right end result. The biggest challenge for externals is usually to understand the data and the course of actions of the case company according to a interviewee.

Despite these some mentioned challenges there were many positive experiences and interviewees were overall satisfied for the collaboration with externals: One interviewee (2) explains that as providing SAP Fiori development is their [externals] business, the collaboration process is in order in their end and there has not been any major problems or challenges. Three other interviewees are agree: The chosen external has had previous experience with case company and therefore knows already about the case companies processes at least to some extent and they have been able to suggest solutions that has not been considered internally or even came up, and they have been very content. In the case (5) where they applied quite agile process and iteratively tested during the process, they had also done internally a lot of the planning and preliminary work, as much as they could, and only the things they could not solve or plan themselves were left to the external developers to solve: This was perceived as a good way of working according to the interviewee, to put effort in to the planning and designing internally as much as possible.

The challenges within business collaboration revolve around the number of business organisations and the different needs of each organisation and team that should be met. One interviewee explains that there might be situations where some organisation is against certain development when they see it as a negative effect to their organisation but adds that it is mainly a change management issue. Following challenges were mentioned by the interviewees: There is no common model among business and different organisations want different things from the application. One interviewee also states that there is no possibility to organize such model that would suite everyone, and then adds that it would be possible but then the budget and resources would be very different from what they have had. There has also been some contrary between decision makers and the local end users according to one interviewee in that sense that there has been change resistance within end users when there has not been functionalities that they would need, and the decision makers have been convinced that the solution answers the problem. Another challenge among business is related to the requirements during development according to one interviewee: As the current development model is the customised waterfall model, there is little room for changing the design. It is a challenge if some requirements arise after the external vendor has calculated their proposal based on a specific functional design, and application might be already ready or even tested. The interviewee underlines the need for communicating a date for design freeze to ensure this does not happen, after which only a heavy reason for changing the design would be considered.

Otherwise business collaboration was experienced as good and the collaboration mainly worked according to all interviewees. Within business there is daily collaboration anyway and usually always someone representing the business end user has been involved in the development process which has been good. Feedback gathering has been quite effortless according to a interviewee as they have direct contact to the teams.

Lastly the collaboration aspects with IM are presented. In addition, thoughts of the role of IM in the development process is discussed in more detail as it was seen weakest of the three. According to one interviewee just to get right accesses to externals took two to three months and they are not sure whether the reason for it was that in IM they did not know what accesses they need for externals or for some other reason, but some problems there were. Other interviewee (5) comments that they have noticed within business that development moves forward more smoothly if IM is left out but adds that this feels like subject dependent as in some cases there is good support and in other not. IM is perceived as more of a delay element than a support function. One interviewee describes that they had some expectations about the IM support that was not realized: The expectations was that IM would give a lot of administrative support on what kind of things can be developed and done, and how. They wondered whether there is a gap in understanding and is there an expectation on IM side that business knows what they need in order to do developments. One interviewee explains that within their case in IM there is differences on a person level on the understanding where their application is used and why it is needed. In one other case they had discussion with IM at the beginning about the solution and the interviewees suspects that in IM the thought might have been that the standard application can be utilized and will be utilized more than what happened in the reality. In another case where they had discussions with IM, and IM had questioned the need for development but in the end mutual understanding was achieved that need was necessary, also in the future with SAP S/4HANA.

So at least in two cases the experience has been that they have had discussions with IM about what could be developed and how. One interviewee comment that the collaboration has gone forward a lot. Another interviewee explains that business has quite accurately the knowledge of what is needed and IM has knowledge of what kind of solutions there are, but in their case and environment the processes and models are quite far developed, so the solutions do not necessarily fit directly to the need. One interviewee has experienced that there has been good guidance from IM to where to contact or how some aspect should be considered.

There were positive experiences as well, and it was underlined by few that the collaboration works with all stakeholders, but from the three IM was the weakest link from interviewees' perspective. From one interviewee (5) there was no criticism towards IM and for the analytics part there had been a lot of help from an IM team. One other interviewee (2) also said the collaboration went fluently and suspected that partly it might be because they have all

worked for so long in the company. One interviewee (1) described that it took some effort to find the correct contacts in IM as they started few years ago within these developments. They added that it is not that big a deal in the end but for a beginner it takes some time to learn and find out these things (Interviewee 1). In one case (5) there was no specific help or support needed from IM for the build of Fiori application because there was the needed knowledge within the business about their tools (Interviewee 5). It can be concluded that between business and IM they are missing active collaboration and understanding and there is room for improvement.

When asked about collaboration between different stakeholders (business, IM, and externals) during the development process and of ideas and thoughts for improvement and hopes for future, many ideas were brought up. When specifically asked about the role of IM, collaboration experiences with IM and how the business interviewees see its role, it becomes quite clear that the role of IM could be clearer and developed so that there is internal knowledge for solution proposals and for the coordination and administration of different applications. One interviewee (1) was not sure what is the value of IM in the development process and thinks that either they did not know how to ask for support, or they did not just get any. Therefore, the role of IM has been very invisible for this interviewee but says that the reason might be because they have been in development projects only for a short time (Interviewee 1). During the development process nobody validated their work except for the latest enhancement which was caused because of the development ramp down, when they got comments from IM that what would be ok to do and what necessarily was not reasonable to do. IM could have more active role according to few interviewees and as one suggests that they could support by suggesting different solutions and models how the solutions could be developed and taken forward as well as handle the administrative part as now business needed to know for example what was needed for the externals which is quite far from their knowledge area. There is a aspiration that IM would be able to support more and one suggests that it would be good if there were Fiori specialists internally to help out in the developments, what is needed for the developments and what kind of development can be done: One interviewee explains that today it feels like everything has to be done by externals, even the smallest changes.

6.4 Service design and prototyping methods

When interviewees were asked about methods business has used for prototyping the Fiori applications, it was found out, there were many prototyping methods used during development process for (rapid) prototyping. Methods such as drawing or writing on post it's and paper, using power point for creating drafts of screenshots by using boxes and arrows and making those clickable. External vendors have provided web-based demo versions based on

those rapid prototypes. These demo versions have been demoed to some users, such as key users or a representative group of users for visually explaining the solutions and for gathering feedback of the demo version. In at least one case SAP's standard applications were used as a base for the application but as standard applications were not applicable out of the box, they required customisation to be done: Editing the prototypes was done according to received feedback from users by cut and paste style according to one interviewee (1). Three out of five had used SAP Build tool for prototyping purposes on different ways: In two cases the tool was used internally, by a team member or the interviewee themselves, and in the third case the tool was used by external in physical or virtual meeting with the interviewee, where the interviewee explained how the solution should work and look like and the external designed the prototypes at the same time.

Service design methods like personas or journey map did not come up during interviews. During interviews did not come up that there would have been a service design ideation workshops held with cross-functional persons in the beginning of the development process.

6.5 SAP Build and design-led development

Next presented results are experiences and thoughts about design-led development model and SAP Build tool. Three out of five had some experience of using SAP Build during their development process (Interviewees 2, 4 and 5). One interviewee (1) had not used or heard of the tool. One (3) had heard and read about the tool and thought it seemed handy during their case, but they had already done some prototypes and the interviewee had thought that maybe it could be used next time. In same case externals had provided a web-based demo version but the interviewee was not aware how they had done the prototypes, with SAP Build or some other way. Although SAP Build has many elements and functionalities for the end-to-end development process, not all of them were available in the free trial version which was used by all the three interviewees, and it was not used for other parts than prototyping the screens. At the time the tool was not very stable, at least the free trial version, one interviewee says and wonders that it might in better shape today. Other interviewee who had designed the prototypes themselves said that it was simple to use and successful without any instructions. The interviewees who had not used the tool thought that if the tool would be easy to learn and not require lot of training, it might be good for development purposes (Interviewees 1 and 3. For prototyping purposes, the tool was easy to learn but whether it would be simple to use for the whole development process and when utilizing all the functionalities in the licenced version, there is no answer. The tool was used by all three for the prototyping purposes and not so much for other parts in the development (Interviewees 2, 4 and 5). Therefore, the

development process did not follow the design-led development process with SAP Build.

After explaining what design-led development model is, two commented that it would be a better way compared to making a long Word-document in text format for the externals and then wait for some time to see what they have developed. In one case the design-led development was familiar as a term and it was quite close to how their development process had gone, without realising they were doing it that way. Third interviewee comments that more iterative way of working is more functional and that it involves the end users from the start to comment on the user interface side and applications functionalities. Two interviewees highlight the importance of the back-end side, as in order to have a workable user interface the logic behind it needs to be thought through and there lies many possibilities there. It is important to remember that it is not all about the user interface and its design although it is as well an important part of the application.

What was seen as a risk by one interviewee in the iterative model was that if the base work, the ideation and planning, has not been well done in the beginning as you can change the plan all the time, it would require dedicated resources who are involved from start to finish of the development. Also as previously mentioned, how the budget is calculated. In waterfall model the budget has been calculated so that the assumption is not to have everything ready for gate two, but you have a restricted scope and the plan is to do enhancements to it in the future separately. The iterative way of working is seen as the future and as better than the waterfall model, however. One person pondered that iterative design-led process would be more meaningful for the users. The problem with waterfall model is the early definition of the functionalities and the design, as usually new requirements arise during the development.

When asked thoughts about SAP Build tool, interviewees mentioned the possibility to visualize the end-user's idea from text form before moving it to development. In some cases, the end users have not known what their wish means in practise and there has been unpleasant surprises when the requirements have been implemented. According to one interviewee the experience was that it is particularly good tool in the beginning of the process for giving the proof of concept (Interviewee 2). Another interviewee also thought it was good tool for selling the idea forward. According to one of the interviewees (1), by visualising the idea to a prototype you could do the disappointment or excitement for the users already at the early stage (instead of users giving input in the requirements phase and then waiting for months in silence before the development is done to see the result). It was experienced by two interviewees that it was easier for end users to implement the tool if they had been a part of the development process.

When asked whether interviewee would be interested to use SAP Build kind of tool for SAP Fiori development process, all were eager to try this kind of tool and saw at least value of trying it out before ruling it out. As advantages of

SAP Build was mentioned that especially if it saves time somewhere else, if it is easy to use, if the prototype could be moved directly from the tool to development, fastens the development chain. As a challenge and disadvantages with SAP Build was mentioned the lack of time, as there is no limitless time to devote to one project. Also, if the prototype could not be used as a base and moved to development straight, it would not bring much value. Also, for project management and budgeting purposes some kind of design freeze was mentioned as needed according to one of the interviewees (2) as externals have made their service offering based on certain scope and somehow this should be considered with the iterative approach. When asked whether and end-user could use this tool themselves, many disagreed. End user can not invest their time to training the use of the SAP Build tool and use time to developments as it would take too much time from their daily work. All agreed that it is important that users are involved in the process and to get their feedback and requirements. It was seen that within business support functions that focus on developing the processes, this kind of tool could be useful and bring value especially if it is easy to use and creates time savings somewhere else. All agreed that the redeeming feature is that you can visualize the idea from text format and get relatively concrete idea of the look and feel and user experience quite early in the process. These prototypes can be then shared with different stakeholders, even if it might change, throughout the development process. Two mention that if there would be internal developers or internal resources for developing Fiori applications, this tool could be useful, and if the process would be refined and optimised around the tool. However, external use will be needed in the future as well, and the externals might have their own refined development process with some other tool, when it might be more suitable to follow their process for the development part, one interviewee (2) comments. One interviewee ponders that for a smaller application the tool worked well in their case but is not sure whether that is the most efficient way of working.

The challenges and cons with SAP Build were related to the resource allocation and schedule issues. Few pondered whether end users will prioritize the development and give their input in timely matter and whether they have possibilities to comment. One wondered that end user's enthusiasm towards development probably would depend on their role and attitude towards these kinds of developments. Another interviewee (3) thought that it is good to have feedback from end users but sometimes they prioritize some element more than it would actually have in the bigger picture. If design-led development approach would be used, firstly the roles should be clearly defined, according to an interviewee, and secondly, then it should be possible to allocate dedicated resources for the development.

6.6 Future development

One of the key things discussed in the interviews was collaboration between different stakeholders. Overall view regarding collaboration between all stakeholders was experienced as good within SAP Fiori developments. The role of IM however was perceived as most weak of the three stakeholders as for some the role was not quite clear and some suspected if it was because they did not know to ask for support. Regarding future developments there were hopes that IM would take more of an active role especially in supporting in the beginning of the development so that there would be internal knowhow in the company and there could be less use of externals for the beginning especially. This or having a dedicated resource might also help with the schedule and moving the development forward, as externals are heavily booked and do not have resources to allocate. Internal SAP Fiori knowledge in IM would be "luxurious" as one interviewee (5) said. The role of IM could be more active by providing recommendations for example on how the idea could be build or can it be developed, is there already some similar solution available, taking the administrative part to follow up if any developments are overlapping, et cetera. Today as there is no internal specialists as such, business needs to go to externals right at the beginning to find out what kind of possibilities there are, how the solution would affect, and how much it would cost. However, few interviewees had had internal discussion with IM in the beginning of the process.

What was also mentioned as missing or what could be enhanced in the development process was the need for governance or some sort of administration of different applications which should be done by someone to avoid too similar applications and unnecessary or overlapping development in the case company. As one interviewee explains, developing a Fiori application is sort of a compromise between different stakeholders. There are many businesses in the case company and between these there are different interests and needs that need to be considered. This is also one reason why end users could not take the lead in developments.

As all interviewees mention, the business users do not have time to draft their idea or to learn the used of development tools nor take bigger role in the development. As one interviewee (1) said, "they have time to send that one e-mail about their idea, --- and maybe as short Skype call has been held in addition." Lack of time but also the complexity of developing a Fiori application are reasons why all interviewees stated that end users should not develop these applications themselves. How end users could be involved in the future? All interviewees agreed it is important to have end users in the process as they are the persons operating with the developed solution. Two of the interviewees (Interviewee 1 and 2) mention the use or possibility to use of key users or key user concept in the development process. Key users are a chosen group of users who represent the end users. In first case, key users have been

used throughout the project and in testing (Interviewee 2). The other case mentions it as a possibility the use of key user concept to engage certain end users to the development process (Interviewee 1).

Other ideas for the future development was introduction trainings or information sessions about SAP Fiori developments in the case company where the internal guidelines to Fiori developments would be gone through. One interviewee explains that there is quite a lot of responsibility in the business which is fine for them but if someone would not have 15 years of SAP experience, they would be quite lost. It was unclear what is the development model for Fiori applications and what are the guidelines. Now as the new S/4HANA environment is coming in near future it would be good to have clear instructions and models for it right from the beginning. One interviewee (2) mentioned WeBuy as a bottle neck, where the project could be frozen for weeks as proposal waits for approval in WeBuy. Another mentioned bottle neck was money and budget.

For the challenge about budgeting for the development, the need for more specific scoping in the beginning and sticking with it is suggested as it is difficult from the project management perspective to not have clearly defined scope. Second idea is saving a certain amount in the budget for requirements that come after user validation for example, as they might bring lot of value to the process. Earlier has been also mentioned some sort of design freeze after which only a critical new requirement could be considered. One interviewee concludes that it is difficult to find a solution or a model that is functioning from every point of view.

6.7 Research limitations

Few research limitations should be mentioned. Firstly, the sampling size of the study is limited as interview sampling size was five. Secondly, the researcher's relationship as a part-time employee in case company can be seen as affecting to the objectivity of the study. Also, not all research methods planned to be used were realizable (observation and field notes) because of the changes in the global environment caused by COVID-19 situation and its affects to development projects. Because of the restrictions implemented because of global pandemic, interviews were held and conducted virtually instead of meeting interviewees face-to-face. This also might have affected the interview environment. It can be considered, what is the difference between physical and virtual interview setting and how it can change the interview situation and interviewees mood, attitude, or distractions at interviewees location or environment among other possible factors.

The conventional approach to content analysis is limited in both theory development and description of the lived experience, because both sampling and analysis procedures make the theoretical relationship between concepts difficult to infer from findings. (Hsieh & Shannon, 2005). There are some

challenges when it comes to conventional content analysis approach that might affect to the credibility of the research (Hsieh & Shannon, 2005). First challenge is failing to develop a complete understanding of the context and therefore failing to identify key categories which can result that findings do not accurately represent the data. To establish credibility, following activities can be taken: peer debriefing, prolonged engagement, persistent observation, triangulation, negative case analysis, referential adequacy, and member check (Lincoln & Guba, 1985; Manning, 1997; according to Hsieh & Shannon, 2005). Second challenge is the similarity to other qualitative methods as it can be confused with other methods such as grounded theory. (Hsieh & Shannon, 2005).

7 DISCUSSION

This chapter concludes the research. With a conventional approach to content analysis, relevant theories or other research findings are addressed in the discussion section of the study (Hsieh & Shannon, 2005). In this discussion chapter the findings are compared to theory. The results from conventional content analysis are compared with literature review to highlight similarities and differences between development process models and to examine if some of the challenges or ideas could be answered with service design approach elements. According to conventional content analysis, the discussion includes a summary of how the findings from study contribute to knowledge in the area of interest and suggestions for practice, teaching, and future research are presented (Hsieh & Shannon, 2005, 1279).

7.1 Summary

The literature review described and compared three approaches to application development: Plan-driven, agile, and service design, and therefore answering for the first research question. Between agile and service design approaches to development there are not that many differences and they are more alike than plan-driven approach. Plan-driven approach is suitable for larger teams and projects (Royce, 1987), and as planning ahead is an important factor, the end result is often known before starting the development process at least to some extent (Abrahamsson et al., 2002). In agile and design-led approaches the end result can be unknown (Abrahamsson et al., 2002): These approaches could be then more suitable to development cases where there is not that clear image of the solution, and there is possibility for ideation and prototyping. Both agile and service design approach utilize prototyping (Cadle & Yates, 2009; Abrahamsson et al., 2002; SAP, 2020e). Prototyping can also be used in plan-driven models (Abrahamsson et al., 2002) but is not as common. In case company where the development model is customised waterfall model,

prototyping has been used to visualize the plan to different stakeholders. In the interview results it was evident that there is a desire for more iterative process according to the study interviewees. Therefore for at least custom SAP Fiori application development, where the project is smaller, this might be a good approach as the end solution is not as well-known necessarily. It needs to be considered if these approaches are manageable in a larger scale when moving to SAP S/4HANA and if standard applications are not sufficient as out of the box solutions. Also, if some areas or the whole process would be changed, what should be then considered is the effort in transformation from plan-driven to iterative - agile or service design - methods: It is a change process that requires careful planning, implementation and anchoring to organisation's development culture (Marttiin, 2020). With the current customised waterfall development model, the change to new model or even few changes can be difficult to implement as for example new way of working requires a lot of effort. The need for application customisation creates more work and requires more resources. If the idea in the case company is to implement the standard applications, the need for heavy development is not as big. However, business interviewees were suspicious whether the standard applications will be suitable for their processes. In the cases discussed in this research the applications have been new development projects where in one case a standard application was used as a base, but it was not utilized much in the end. For new development projects where the end result is not entirely clear and new possibilities are wanted to be explored, service design approach elements could be involved as its methods can be used for ideation for example.

Next the second research question is discussed. Scope, time and cost issues constrain each development project in different ways (Atkinson, 1999). Since end of 2018 there has been high focus on cost savings and not allowing external consultants: Challenge of cost savings presented by Schutte (2019) is still true which comes evident of the budgeting challenges that are faced, the budget and the developments are done with a minimal budget. External development is expensive according to interview results, and many thought it would be beneficial to have some knowhow internally in IM, so that there would not be needed to go externals right from the start. This is not only related to budgeting, as internal expertise was also seen as good possibility for the purposes of planning and designing and how to develop the application so that it fits the company environment for example. An interviewee (5) highlighted the need for efforts in the beginning of the development as it would be better to invest more time on that, so that also a clearly defined idea and scope would benefit the collaboration between externals.

When it comes to challenges with enterprise applications, there were numerous presented in the literature review: legacy, custom, and packaged applications, monolithic code not built for easy maintenance, multiple design and execution technologies that need to be integrated, demand for new technology support and customers that won't wait years for a solution, multiple platforms, databases, transaction processors, data entry points, and versions of

the same data as well as incompatible business data (Spratt, 2000; McKeen & Smith, 2002). One interviewee (4) mentioned the need for processing good data quality in order to have good, workable solutions. There have been some concerns regarding the quality of data and quality of code.

One of the main topics discussed with interviewees was collaboration between different stakeholders during development process. According to Schutte (2019) there are collaboration issues regarding SAP developments within the company such as missing active collaboration and understanding between stakeholders as well as no real accountability for external developer performance. One of the results in Schutte's (2019) study was that the process is not supporting the desired collaboration between the stakeholders, and this was also evident in the interview results especially between IM and business. These same challenges came up in the interviews although mainly collaboration was perceived as good between different stakeholders within SAP Fiori developments. The role of IM however was perceived as most weak of the three stakeholders as for some the role was not quite clear and some suspected if it was because they did not know to ask for support. Regarding future developments there were hopes that IM would take more of an active role especially in supporting in the beginning of the development so that there would be internal knowhow in the company and there could be less use of externals for the beginning especially. Providing recommendations for example on how the idea could be build or can it be developed, is there already some similar solution available, taking the administrative part to follow up if any developments are overlapping, et cetera. As appose to Schutte (2019), within SAP Fiori development there was more accountability for external developers. However, there was some suspicion by one interviewee (1) of the quality of the code as the testing had been quite laborious and almost every time something had been changed, the whole solution broke.

For collaboration challenges perhaps service design methods and process could be applied. Applying methods like in the ideation phase would require the different stakeholders to be involved right from the beginning of the development. For example, in the future prototyping could be still used as there were many prototyping methods used by business for ideating and visualising the solution and they are familiar with prototyping. (Rapid) Prototyping was familiar for the interviewees according to answers related to the development process and prototyping methods. Using post-it notes or drawing and drafting to paper screenshots were often used methods as well as using power point for more detailed and more invested prototypes. Screenshots created by Microsoft PowerPoint were done by adding boxes and arrows and these were made clickable in later state of the prototyping. Discussions throughout the development were held, usually having one or more end users involved. The prototypes or solutions were demoed for end users or a representative of end user for gathering feedback. Changes to prototypes were made according to feedback. Prototyping with SAP Build tool was perceived as a good way to visualize, explain and demo the solution to different stakeholders in all stages

of the development. Compared to using PowerPoints, SAP Build was perceived as good solution for SAP Fiori prototyping especially as the look and feel and user experience of the application can be seen and tested already at early stage and also as the clickable prototypes were simple to design. For collaboration issues it could be considered if IM would be involved already in the scope and requirement phase, as collaborative teams and process are fundamental for service design approach. The challenge with this is resources: Is there resources in IM to be involved to all SAP Fiori development projects? If there could be internal team who would have this development as their responsibility, as interviewees suggested, then it might be possible. This also would depend on how many SAP Fiori projects there are simultaneously on-going and whether the team would have enough time to be involved in each of them.

According to Jones and Marsden (2006) a good user experience is a sum of several different elements, such as the product should respond to user needs and provide a solid and easy-to-learn user interface, as well as offer pleasant experiences in general. Fiori applications were perceived as easier and faster to learn by users, leading for need for less training. Saving on training costs was one of the mentioned SAP Fiori benefits for business by SAP (2020d). The feedback received from end users about Fiori applications has been mostly positive, but some negative feedback has also been received. Negative feedback has been regarding the screen layout of the applications for example. There has been discernible some change resistance.

According to the research results, SAP Build tool should not be used by end users as they do not have time and they do not have the knowledge on the developments or what is required for other users for example. Requirements for the application come from business and end users are important to have in the process throughout it and their feedback is valuable, but end users do not have time to start prototyping the ideas themselves. This could be done as today in the business support functions that were the focus group for this research. However, users are important to have involved in the development process throughout the process to give their input on the requirements and give feedback based on prototypes. SAP Build tool was perceived as good opportunity According to the interviewees, the benefits of the tool were the possibility to visually present the idea to different stakeholders for proof of concept and to gather feedback early on from users for example. All three interviewees (2,4, and 5) had used the free trial version of the tool; hence they did not have experience of all the functionalities within the licenced version.

One of the main challenges in the development process was collaboration between IM and business, although it must be highlighted that it was perceived as most weak out of the three stakeholders and there are also many areas of success. Many hopes for future development however entailed the wish for better collaboration between IM and business. IM could be involved in the process from the beginning to give their input on what is possible to do in the case company environment for example. IM could contribute in the starting discussions where scope and requirements are in discussions. IM could be also

involved in the prototyping. The challenge with involving IM more in the development process and adding more collaboration between with business and IM is the resourcing because there are multiple development items, ad-hoc requests and firefighting which the resources are assigned to.

This research started with an introduction and was followed by a literature review in which the main concepts of this case study were defined: Application development and service design. First chapter focused on defining application development, and two of its development approaches, plan-driven and agile development approaches. This was followed by defining service design and its principles, design-led development process according to SAP Build tool and the methods used in the process. After defining these main concepts of this research, a comparison and analysis of the challenges and similarities of these concepts was made. The goal was to identify the similarities and differences between the different development approaches and answer the research question: How traditional application development approaches, and service design approach differ from each other? This was done to analyse if the challenges in application development processes could be improved by applying elements of service design approach. The literature review ended to a summary and conclusions of the theory. Third chapter consisted of case study introduction where the case company was introduced. As a part of the case chapter, enterprise resource planning (ERP) system was defined and ERP system providers and products that are relevant to the case research were introduced: SAP, SAP S/4HANA, and SAP Fiori. The fourth chapter described the research methods, data collection and data analysis processes. For data analysis, conventional content analysis was applied according to Hsieh and Shannon (2005). Data analysis was followed by results and discussion. The research ends with key findings and proposals for future research which are addressed next.

7.2 Research questions and key findings

The research questions for literature review and case study were:

1. What kind of differences there are between well-known application development approaches and service design approach?
2. What kind of SAP Fiori development and SAP Build tool experiences business has and what kind of hopes there are for future development?
3. What kind of elements of service design approach could be integrated to SAP Fiori development process to prevent the challenges and address the future wishes?

The key findings are presented next. In this research it was found that between agile and service design approaches to application development there are not that many differences and they are more alike than compared to plan-driven

approach. Service design alike with agile approach is applicable approach especially when the end result is not entirely known, or new ideas are wanted to be discovered. However, these approaches require more dedicated resources than plan-driven approach, as the collaboration requires more input more often.

In the interview results it was evident that there are collaboration challenges between business and IM that could be improved. During the development process the role of information management (IM) organisation was most unclear from business perspective according to interview results and more activity and input was hoped for the future especially for the start of the process. It was unclear for the business what is the development model and guideline for SAP Fiori applications in the case company so these could be clarified. For the future, more active role for IM was hoped for. SAP Build tool was perceived as good tool for proof of concept for different stakeholders and for easy visualisation of the solution at an early stage. It was not however perceived as a tool that a business end user could use as they inter alia do not have time for it. But the value was seen in the tool if it would be used by business support functions that focus on developing their tools and processes. The free version of the tool was used so not all of the functionalities were available and it was mainly used for the prototyping purposes.

There are some challenges in the SAP Fiori application development and some suggestions were made about service design approach elements that could be included into the process to prevent these challenges in the future. As one of the main challenges were collaboration between IM and business, IM could be involved in the process from the beginning to give their input on what is possible to do in the case company environment for example. IM could contribute in the starting discussions where scope and requirements are in discussions where service design methods could be utilized for especially ideation and planning. Secondly, IM could be also involved in the prototyping to give their point of view of how the solution would fit to the case company environment. The challenge however with involving IM and collaborating more in the development process is the resourcing. It needs to be considered if there are enough resources in IM to contribute in this manner. With these results, the third research question was answered.

This case study teaches to the similar research processes that service design approach could be studied more in the context of application development. The research results can be utilized in other companies as a reference for areas of improvement for future development as many other companies face similar challenges in their development processes. For future research it could be further studied how the integration of service design to plan-driven application development process could be realized in a globally distributed company. Change from SAP ERP to SAP S/4HANA is concerning a number of companies, and it could be studied how the new SAP Fiori user interface is implemented and how the users adopt it.

REFERENCES

- Abrahamsson, P., Salo, O., Ronkainen, J. & Warsta, J. (2002). Agile software development methods: Review and analysis, VTT publication 478, Espoo, Finland, 107.
- Ambler, S. W. (2002). Duking It Out Reloaded. *SOFTWARE DEVELOPMENT-SAN FRANCISCO-*, 10(11), 47-50.
- Atkinson, R. (1999). Project management: cost, time and quality, two best guesses and a phenomenon, its time to accept other success criteria. *International journal of project management*, 17 (6), 337-342.
- Benbasat, I., Goldstein, D. K., & Mead, M. (1987). The case research strategy in studies of information systems. *MIS quarterly*, 369-386.
- Birrer, E.T. Frameworks in the financial engineering domain: An experience report. In *Proceedings of ECOOP '93 Proceedings, Lecture Notes in Computer Science nr. 707*, Springer-Verlag, 1993.
- Boehm, B. (2002). "Get Ready For The Agile Methods, With Care." *Computer* 35(1): 64-69.
- Böder, J., Gröne, B., Joshi, S., Karthik, S. R., Kleis, W., & Kumar, S. (2013). *The Architecture of SAP ERP – Understanding how successful software works*. Tredition GmbH. Hamburg.
- Cadle, J., & Yeates, D. (Eds.). (2009). *Project management for information systems*. Pearson education.
- Chesbrough, H., & Spohrer, J. (2006). A research manifesto for services science. *Communications of the ACM*, 49(7), 35-40.
- Conallen, J. (1999). Modeling Web application architectures with UML. *Communications of the ACM*, 42(10), 63-70.
- Constantine, L. L., & Lockwood, L. A. (2002). Usage-centered engineering for web applications. *IEEE software*, 19(2), 42-50.
- Darke, P., Shanks, G., & Broadbent, M. (1998). Successfully completing case study research: combining rigour, relevance and pragmatism. *Information systems journal*, 8(4), 273-289.
- Dubberly, H., Evenson, S., and Robinson, R. (2008). The Analysis-Synthesis Bridge Model, *Interactions*, 15(2):57-61.

- Edvardsson, B., Gustafsson, A., Sandén, B., & Johnson, M. D. (2000). New service development and innovation in the new economy. *Studentlitteratur*.
- Esteves, J., & Pastor, J. (2001). Enterprise resource planning systems research: an annotated bibliography. *Communications of the association for information systems*, 7(1), 8.
- Evenson, S. (2008). "A Designer's View of SSME," in *Service Science, Management and Engineering: Education for the 21st Century*, Bill Hefley and Wendy Murphy, eds. New York, NY: Springer, 25-30.
- Fawcett, E. (2002). Extreme programming vs. interaction design. FTP Online.
- Fayad, M. E., & Hamu, D. S. (1997). Object-oriented enterprise frameworks: Make vs. buy decisions and guidelines for selection. *Communications of the ACM*.
- Fayad, M., & Schmidt, D. C. (1997). Object-oriented application frameworks. *Communications of the ACM*, 40(10), 32-38.
- Heuristic. (n.d.). In ICIS . (2020). Track Descriptions. Conference of the Association for Information Systems (AIS). Retrieved 15.09.2020 from <https://icis2020.aisconferences.org/>
- Heuristic. (n.d.). In SAP Help Portal website. (2020). Website: Explore SAP Fiori. Retrieved 11.9.2020 from https://help.sap.com/viewer/f42cbf488a3c4f18a570b20c57b77cfa/5_OVEVIEW/en-US/f009e97dbcc94463aabfac350b09fcf5.html
- Heuristic. (n.d.). In SAP website. (2020a). Website: What is SAP S/4HANA? Retrieved 02.06.2020 from <https://www.sap.com/products/s4hana-erp.html>
- Heuristic. (n.d.). In SAP website. (2020b). Website: SAP HANA and S/4HANA. A simple guide. Retrieved 09.06.2020 from <https://blogs.sap.com/2020/06/04/sap-hana-and-s-4-hana-a-simpleguide/>
- Heuristic. (n.d.). In SAP website. (2020c). Website: Design. Develop. Deliver. A modern UX for every device. And every user. Retrieved 11.09.2020 from <https://www.sap.com/products/fiori.html>
- Heuristic. (n.d.). In SAP website. (2020d). Website: SAP Fiori Design Guidelines. Retrieved 06.05.202 from <https://experience.sap.com/fiori-design/>
- Heuristic. (n.d.). In SAP website. (2020e). Website: Learn Design-Led Development. Retrieved 27.02.2020 from <https://www.build.me/learning>

- Heuristic. (n.d.). In World Wide Web Consortium website. (2020). Website: W3C Accessibility Standards Overview. Retrieved 04.05.2020 from <https://www.w3.org/WAI/standards-guidelines/>
- Highsmith, J. A., & Highsmith, J. (2002). Agile software development ecosystems. Addison-Wesley Professional.
- Hirsjärvi, S., Remes, P. & Sajavaara, P. (2009). Tutki ja kirjoita. (In Finnish). (15. uud. painos). Helsinki: Tammi.
- Hollins, G., & Hollins, B. (1991). Total Design: Managing the design process in the service sector (p. 172). London: Pitman.
- Holmlid, S. (2007). Interaction design and service design: Expanding a comparison of design disciplines. Nordes, (2).
- Holmlid, S., & Evenson, S. (2008). Bringing service design to service sciences, management and engineering. In Service science, management and engineering education for the 21st century (341-345). Springer, Boston, MA.
- Hsieh, H. F., & Shannon, S. E. (2005). Three approaches to qualitative content analysis. *Qualitative health research*, 15(9), 1277-1288.
- International Organization for Standardization. (2017). Systems and software engineering – Vocabulary. (ISO/IEC/IEEE 24765:2017 (en)). Retrieved from <https://www.iso.org/obp/ui/#iso:std:iso-iec-ieee:24765:ed-2:v1:en>
- Jacobs, F. R., & Weston Jr, F. C. (2007). Enterprise resource planning (ERP) – A brief history. *Journal of operations management*, 25(2), 357-363.
- Jones, M., & Marsden, G. (2006). Mobile interaction design. John Wiley & Sons.
- Kimbell, L. (2009). Insights from Service Design Practice. 8th European Academy of Design Conference, 249-253. Aberdeen.
- Laitinen, M., Fayad, M., Schmidt, D., & Johnson, R. (1999). Object-oriented application frameworks: Problems and perspectives. Wiley, 9.
- Mager, B. (2008). Service Design. In M. Erlhoff, & T. Marshall (Eds.), *Design Dictionary: Perspectives on Design Terminology*, 354-356. Basel: Birkhäuser.
- Martiin, P. (2020). Agile methods & agile adoption as IT enabled change. Lecture 5. Course. TJTSM53 Project and Change Management. University of Jyväskylä. Jyväskylä.

- McKeen, J. D., & Smith, H. A. (2002). New developments in practice II: enterprise application integration. *Communications of the Association for Information Systems*, 8(1), 31.
- Meroni, A., & Sangiorgi, D. (2011). *Design for services*. Gower Publishing, Ltd.
- Myers, M. D., & Newman, M. (2007). The qualitative interview in IS research: Examining the craft. *Information and organization*, 17(1), 2-26.
- Myers, M.D. (ed.) (1998) *Qualitative Research in Information Systems*. Retrieved from <http://www.auckland.ac.nz/msis/isworld/>
- Neale, M.R., & Corkindale, D.R. (1998). Codeveloping products: Involving customers earlier and more deeply. *Long Range Planning*, 31, 3 , 418-425.
- Panorama Consulting Group. (2020). *The 2020 ERP Report*. Retrieved from <https://www.panorama-consulting.com/resource-center/2020-erp-report/>
- Patrício, L., Fisk, R. P., Falcão e Cunha, J., & Constantine, L. (2011). Multilevel service design: from customer value constellation to service experience blueprinting. *Journal of service Research*, 14(2), 180-200.
- Rizzo, F. (2010). Co-design versus user centred design: Framing the differences. Guerrini L, ed. *Notes on Design Doctoral Research*, Chapter 12 (Franco Angeli Editore, Milan).
- Royce, W. W. (1987). Managing the development of large software systems: concepts and techniques. In *Proceedings of the 9th international conference on Software Engineering*, 328-338.
- Sangiorgi, D. (2009). Building a framework for service design research. In 8th European Academy of Design conference, 415-420. GBR.
- Segelström, F. (2010). *Visualisations in Service Design*. Linköping, Sweden: Linköping University.
- Shostack, G. L. (1984). "Designing Services that Deliver." *Harvard Business Review*.
- Sol, H. G. (1983). A feature analysis of information systems design methodologies: Methodological considerations. *Information systems design methodologies: A feature analysis*. T. W. Olle, H. G. Sol and C. J. Tully. Amsterdam, Elsevier: 1-8.
- Song, X. & Osterweil, L. J. (1991). Comparing design methodologies through process modeling. 1st International Conference on Software Process, Los Alamitos, Calif., IEEE CS Press.

- Stickdorn, M., Hormess, M. E., Lawrence, A., & Schneider, J. (2018). *This is service design doing: applying service design thinking in the real world.* " O'Reilly Media, Inc."
- Szirtes, T., & Rivlin, A. (2015). *Implementing SAP Fiori Launchpad.*
- Venkatesh, V., Thong, J. Y., Chan, F. K., & Hu, P. J. (2016). Managing citizens' uncertainty in e-government services: The mediating and moderating roles of transparency and trust. *Information systems research*, 27(1), 87-111.
- Wetter-Edman, K., Sangiorgi, D., Edvardsson, B., Holmlid, S., Grönroos, C., & Mattelmäki, T. (2014). Design for value co-creation: Exploring synergies between design for service and service logic. *Service Science*, 6(2), 106-121.
- Vosough, Z., Walter, M., Rode, J., Hesse, S., & Groh, R. (2016). Having fun with customers: Lessons learned from an agile development of a business software. In *Proceedings of 9th Nordic Conference on Human-Computer Interaction.*

APPENDIX 1 INTERVIEW QUESTIONS

First section: For the start

- Would you tell what is your job description and how long have you worked in the case company?
- How much you operate with SAP system in your daily work and what is your overall opinion of the usage of the system?
- What is your overall opinion and experience of SAP Fiori applications?
- In how many SAP Fiori application development project have you been involved in?

Second section: SAP Fiori development today

- Have you used SAP Build tool for developing and designing SAP Fiori application?
- Would you tell how the SAP Fiori project(s) went from beginning to end?
- How long did the project(s) last?
- Was there problems and challenges during the process? What kind?
- How about, what worked and succeed during the process?
- How did collaboration work between different stakeholders (Business, IM, externals)?
- How did communication work between different stakeholders (Business, IM, externals)?
- What do you think about the final outcome? Are you happy of the usability of the application?
- If you have used SAP Build, how would you compare the processes between each other? Would you use happily SAP Build again?

Third section: SAP Fiori development in the future

- What do you wish would be changed in the SAP Fiori development process and do you have ideas how things could be changed?
- What kind of hopes do you have regarding SAP Fiori development in the future?
- Would you like that business would have more possibilities to impact the applications look, usability and user experience, and designing these aspects? In the interviews it was clear that business has a lot of impact

Fourth section: Design led development with SAP Build

- What do you think about this development model?
- What do you think about different (design) tools?

- What would you think if business side would have more responsibility regarding designing the look and content?
- Business plans and tests prototypes until it answers to needs. After this prototype moves to development.
- Are you interested (as an end user) to do more work related to application design? The interviewees were not end users as thought before the interviews were held

Fifth section: For the end

- Would you still like to tell something regarding the theme that was not considered in the interview?